# Commodore INFO

**Opnieuw de C-128**

**64 Machinetaal**

**Uitslag Idee 1990**

ONAFHANKELIJK BLAD VOOR COMMODORE GEBRUIKERS PRIJS ƒ 5,75/ Bfr 120 JAARGANG 2, 1985, No. 3

## LISTINGS

- KLEURSHOW
- VIC 20 HOGER-LAGER
- RADEN MAAR VIC 20
- VIC 20 LOTUS
- LOTTO
- WOESTIJNRACE
- SPELRECORD 64
- CRAZY SPRITE 64
- C-16 DATABASE




**Softprom**

**Robots**

**Malmberg software**

**C-16 Database**

**In gesprek met Logo**

**C-16 Machinetaal**

**Kortingsbon voor het Grote Commodore-Info Listingboek**

# Redactioneel

*We zijn de opwinding van het beurzenseizoen al weer bijna te boven, al heeft het heel wat zweetdruppeltjes gekost. Een succesvolle Commodore-Info op 14 september, de wat ons betreft redelijk geslaagde PCM Show in Utrecht en de HCC dagen komen er nog aan (22 en 23 november in de Jaarbeurs). Nederland blijkt zich ook dit najaar weer massaal in de computerspullen te gaan steken, en dat juichen we van harte toe.*

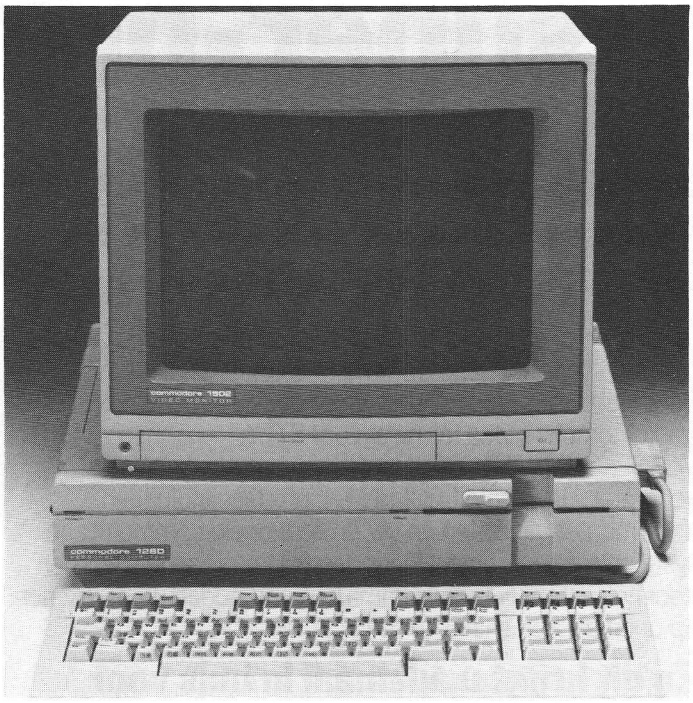
*Op de PCM bleek de Nederlandse Microcomputer Trofee voor het beste Commodore software-pakket te gaan naar Radarsoft voor de adventure* **EINDELOOS**, *een zeer juiste keus. Wel hopen we, dat men het draagvlak van deze verkiezing volgend jaar wat breder maakt en bijvoorbeeld ook een categorie* **Beste Exportprodukt** *toevoegt.*

*Dat Commodore niet zo blij was met ons verhaal in het vorige nummer over de C-128, dat kunnen de lezers begrijpen. Volgens Commodore hebben we echter een van de vele niet-volmaakte C-128 versies getest en zou er dus wel hoop zijn voor de "opvolger". Een discussie daarover gaan we niet uit de weg, zolang onze onafhankelijkheid voorop blijft staan. We hebben de 128 opnieuw bekeken en U leest elders in dit nummer een nader oordeel. Tot een definitieve uitspraak willen we nog niet komen, daarvoor zijn er ook nu nog teveel onzekerheden. Wat ons betreft krijgt de C-128 echt wel een faire kans, maar van loze promotiepraat blijft de lezer verschoond.*

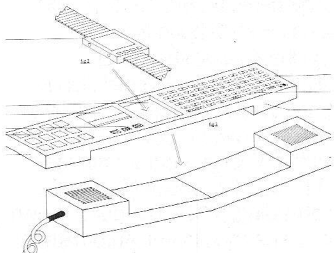
*Vanaf deze plaats veel dank voor de overweldigende inzendingen voor onze Idee- en Programmeerwedstrijd. We zijn er in kwantiteit en kwaliteit bijna onder bedolven, maar zeker uit de Ideëen voor de* **Computer Anno 1990** blijkt wel, dat we qua creativiteit in de Benelux nog heel wat in huis hebben.

L.Sala

## Inhoud van dit nummer











# LUC SALA'S DATAKOLOM

**Echte Innovatie**

*Chips kun je gebruiken in vele apparaten en wie zijn wasmachine, hi-fi installatie, TV of autoradio openschroeft, ziet ze zo zitten. Toch is het jammer, dat er buiten het computergebeuren maar zo weinig echte innovatie te bespeuren valt in de toepassing van chips. Want de fabrikanten stoppen gebruiken chips in een televisie, wasmachine of radio in de eerste plaats vanwege de kostenbesparing, het lager gewicht, kleinere storingskans en dus lagere garantiekosten, minder componenten etc. Dat er dan hier en daar een stukje kwaliteitsverbetering bij komt, bijvoorbeeld een fraaier beeld of wat videoprocessing grapjes bij digitale televisie, is meegenomen. De gebruiksvriendelijkheid is wel toegenomen, we hebben tegenwoordig afstandsbediening en de kostprijs van de apparatuur is relatief lager geworden, maar de TV van nu is als produkt niet veel verschillend van de TV van 10 jaar geleden. Hetzelfde geldt voor wasmachines, versterkers en vrijwel alle zgn. "domestic appliances" ofwel huis- tuin- en keukenapparatuur uit de wit- en bruingoedsector. Denk er aan, de Laserdisk techniek is geen kwestie van chips, al is door de chips het produkt wel levensvatbaar en betaalbaar is geworden.*

*In de computers zijn de chips, en dan met name de geheugenchips, wel de oorzaak van stevige veranderingen en is computeren nu wel degelijk verschillend van computeren 5 jaar geleden. Dat is verheugend en hoewel de ontwikkeling de laatste 2 jaar ook niet meer zo snel gaat, zien we in de koppeling van bv. de huiscomputer aan de laser-technieken (CD-ROM) toch wel weer nieuwe perspectieven.*

*Zonder nu al te negatief te willen oordelen over de activiteiten van de grote elektronika-concerns, heb ik toch het gevoel, dat men daar soms de echte innovatie uit de weg gaat en liever met de ontdekte technieken de bestaande produkten of produktie goedkoper maakt. Waar zijn de oplossingen voor de milieuproblemen, voor de verkeersveiligheid, de communicatie tussen mensen die elkaars taal niet verstaan, de individuele veiligheid, de identificatie van goederen, onderhouds- en slijtage-waarschuwingen, de elektrische veiligheid van apparatuur etc. etc. Met belangstelling lees ik af en toe persberichten of ga ik naar demonstraties van bv. het CARIN autoinformatie-systeem van Philips, hoor over computertjes in sportschoenen of robots als huisbutler, maar zoiets zie ik nog lang niet als echte praktijkprodukten, het zijn aardigheidjes voor wie geld te veel heeft. Ik denk, dat grote bedrijven ook niet de beste plaatsen zijn om echt iets revolutionairs te ontwikkelen en dat kan ik ook wel toelichten aan de hand van een voorbeeld. Ik ben zelf van huis uit natuurkundige en zo'n 4 jaar geleden had ik iets bedacht, dat naar mijn mening wereldwijd een paar procent van alle elektrische energie zou kunnen besparen. Het kwam erop neer, dat door het inbouwen van relatief eenvoudige schakelingetjes in huishoudelijke apparaten de piekbelasting voor het electriciteitsbedrijf kon worden afgevlakt. Een extra chipje in alle relatief tijdsonafhankelijke apparaten zoals vaatwasser, wasmachine, electrische verwarming of diepvriezer en dan op commando van het electriciteitsbedrijf de zaak op afstand (en met wat random tijdvertraging) in- of uitschakelen, wanneer dat voor een meer evenwichtige stroomafname nodig is. Een simpel codesignaaltje via het net, en hup, er komen wat stroomvreters bij om de zaak te egaliseren of er vallen er een aantal af. Dus minder pieken in het stroomverbruik, daarmee efficiëntere benutting van de centrales en dus besparing.*

## Piekscheren

*In wezen is het nachtstroom-tarief al iets op die weg, maar dat is wel erg grof. Technisch allemaal niet zo ingewikkeld en tegenwoordig zijn de electriciteitsbedrijven ook wel bezig dit met industriële afnemers zo te regelen en zijn er kleinschalige experimenten met bv. het afschakelen van wasmachines.*

*De consument blijft echter een grote energieverspiller en wie zou dus beter een breed besparingsidee naar de consument toe kunnen brengen dan de maker van die huishoudelijke apparaten.*

*Welnu, mijn brieven met plannen aan het management en de laboratoria van de vaderlandse electronikagigant hadden mooi geen effect. De mensen, die zich daar bezighielden met die apparatuur vonden iedere verandering, die uitging boven andere kleurtjes, knopjes of produktiekostenbesparing maar niks, ze hadden de produktie nu al jaren mooi opgelijnd, waarom zouden ze daar iets aan veranderen. Als anderen het zouden gaan doen, ja dan zou men natuurlijk overwegen om ook dergelijke voorzieningen in te bouwen. En meer van dergelijke kul-argumenten, het lijkt er meer op, dat grote organisaties te traag zijn en echte innovatie nauwelijks aankunnen. Ze doen aan basisresearch en hebben de techniek in huis om echt dingen fundamenteel te veranderen, maar de logheid van hun eigen organisatie houdt dat vrijwel altijd tegen.*

*Jammer, van mijn idee, men was er toen niet gevoelig voor. Maar dat geldt ongetwijfeld ook vele andere en betere ideeën van anderen, innovatie en frustratie gaan vaak hand in hand. En wanneer ik toen energiechips was gaan maken, was ik natuurlijk nooit meer aan computerbladen toegekomen.* ●

# nieuws

## Popgroep als videospelletje

Het moest er eens van komen. Na picturedisks, videoclips nu de videospelletjes met bekende popgroepen. **Frankie goes to Hollywood** bijt de spits af met het gelijknamige spel. Het doel is de Pleasure Dome te bereiken. Daarvoor is het noodzakelijk 87000 punten te verwerken via de 60 tijdens het spel gegeven opdrachten. Wat verwarrend van opzet, maar grafisch fraai uitgevoerd. Voor de liefhebber.

## DIA-ARCHIEF C-64

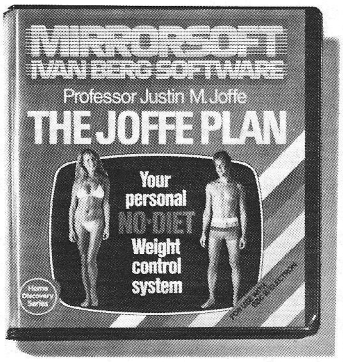
Meer dan 32000 dia's onder maximaal vier kategorieen kunt u kwijt in het dia-diskarchief van H.Vierke. De prijs bedraagt fl. 140,-.
Verder info Mackayware 90, 8014 RC Zwolle.

## SURF-RAPPORT

Het Surf-rapport behandelt niet het jaarverslag van de Nederlandse plankzeilers, maar staat voor Samenwerking Universitaire Rekenfaciliteiten en geeft de opzet voor een stimulering van het computergebruik op de universiteiten en dan vooral op PC gebied.

## Joffe Dieet

Van Mirrorsoft en met Nederlandse handleiding via Aackosoft is er het Dieetloze Joffe




afvalplan. Ja, dat gaat wel om de slanke lijn, de Joffe methode maakt voor iedereen, afhankelijk van zijn levensstijl, een aktieplan om fitter te worden. Uiteindelijk moet u het zelf doen, maar de computer is hierbij wel de raadsman, die het streefgewicht op wetenschappelijk verantwoorde wijze aan u oplegt. We kennen fitness cursussen, die meer kosten dan de *f* 49,50 van **Het Joffe Plan.**

## Bridge

Dit is een Nederlands programma van Bridgesoft uit Naarden, dat belooft ook in het buitenland een succes te worden, het produkt wordt ook in de VS en Engeland op de markt gebracht. Het is meer een programma om te leren bridgen dan om het spel tegen de computer te spelen. De computer bepaald de spellen en de reac-

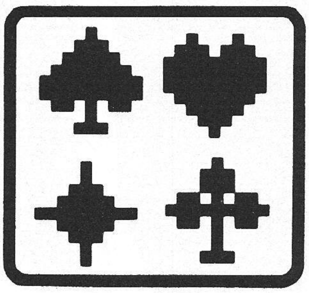
**COMMODORE-INFO** Jaargang 2, no 9, 1985

**Uitgave:** Sala Communications
**Uitgever:** Drs. J.Taverne
**Marketing/Beurzen:** Ing. V.Sala

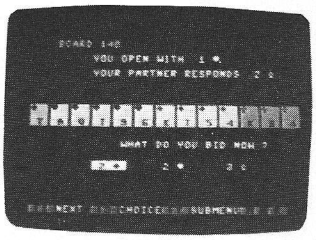
**Redaktieadres:**
Postbus 112, 1260 AC Blaricum
☎ 02152-65695
**Abonnementen en administratie:**
Postbus 5570
1007 AN Amsterdam
Den Texstraat 5a
1017 XW Amsterdam
☎ 020-273198
**Abonnement** *f* 55,- of Bfr. 1100 per jaar (10 nummers).
Betaling op Giro 158591 tnv. SAC/COMMODORE-INFO Blaricum of in België op Bank BBL nr. 310050602562, vermeld SAC/COMMODORE-INFO. Oude nummers à *f* 6,75 alleen bij vooruitbetaling op giro 1585491.

**REDAKTIE**
*Ir. L.Sala* — *hoofdredacteur*
*J. Bodzinga* — *adj. hoofdred.*
*K. van der Vlies* — *reportage*
*B. Tier* — *strip*
*B. van Mierlo/ Ymmot* — *art*
*R. van den Heuvel* — *listings*
*B. Munniksma* — *onderwijs*
*Sj. Bakker* — *machinetaal*
**Redactiesecretariaat**
L. Hooper
J. Emmelot


ISSN: 0169-3085
NDB Zoeterwoude
**Druk:** Verweij, Mijdrecht
**Distributie:**
in Nederland — Betapress/Gilze
in België — AMP/Brussel



ties van de partner en tegenstander, maar de speler kan wel zijn eigen fouten maken.

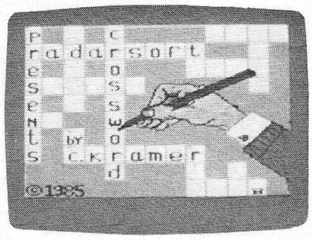



Deze bridgecursus, die werkt met de meestgebruikte biedconventie en dan ook aansluit bij bv. de TV- cursus over Bridge, is te koop bij MCN. Er komen ook vervolgprogramma's, waarin men op basis van zo'n 150 nieuwe spellen zijn bekwaamheden verder kan ontwikkelen. We zagen dit programma en hoewel de graphics vrij klassiek zijn, maakt de gedegen educatieve aanpak van dit programma het de moeite waard.

## Aktuele Microcomputer Techniek

Van Weka kregen we onlangs het PRAKTIJKBOEK Aktuele Microcomputer Techniek toegestuurd en dat betekent dan wel een forse map met plm. 300 pagina's. Qua uitvoering hebben we iets dergelijks nog niet gezien, dit is geen computerboekje meer, maar een compleet naslagwerk, dat lijkt op de losbladige vakliteratuur, die bv. bij grote computers wordt bijgeleverd. Qua uiterlijk en afwerking van zeer hoog niveau en ook de inhoud is niet mis. Men houdt niet op bij algemene aanbevelingen, bij het hoofdstuk over printplaatjes zit er een transparant bij, waarmee men zelf aan de gang kan.
In de 14 delen van het basiswerk (er volgen nog aanvullingen, waarop men zich abonneren kan) komen vele onderwerpen aan de orde. Sommige zijn zeer diepgaand met programmavoorbeelden en veel schema's, andere geven wat oppervlakkiger informatie over bv. nieuwe computers op de markt zoals de 128 en Atari 520. Er is veel "harde" informatie, die dit als naslagwerk heel geschikt maken, zoals lijsten met verkrijgbare processoren, adressen en literatuuroverzichten. Hierbij vinden we, dat men wel wat overspringt van de ene machine naar de andere, zo is de basic- programmeercursus aan de hand van de C-64, maar de machinetaalcursus werkt met de MPS-65 en de literatuurlijst gaat voornamelijk over wat er voor de Spectrum beschikbaar is. Misschien kan men dit in de aanvullingen wat meer richten.
In het algemeen is dit praktijkboek, dat *f* 99,- kost, dat geld meer dan waard voor de serieuze hobbyist, die wat meer aan de gang wil met machinetaal, interfaces en microprocessor- techniek.

## Kruiswoord generator




Radarsoft begeeft zich tegenwoordig ook op andere gebieden dan adventure's (Eindeloos) en educatieve actiespelen (Topografie) en met Kruiswoordgenerator mikt men op de grote groep mensen, die deze vorm van puzzelen beoefenen. Maar het programma gaat ietsje verder dan puzzelen, er blijft een educatief tintje aan, want het vinden van de juiste woorden stimuleert het taalgevoel.

## Rekenmonster

Een waar rekenmonster is de nieuwe Cray-2 Super Computer. Met 2000 MFLOPS (=floating operations per second) klopt deze snelle jongen al het

tot op heden ontwikkelde rekentuig. De Cray-2 is 114 cm hoog bij een diameter van 135 cm. Om de signaaltransport beneden 4,1 nanoseconde te houden is er geen enkele kabelverbinding langer dan 80 cm. Als u er onverhoopt eentje wilt aanschaffen dan komt dat op zo'n 20 miljoen US dollars.

## Amiga nieuws

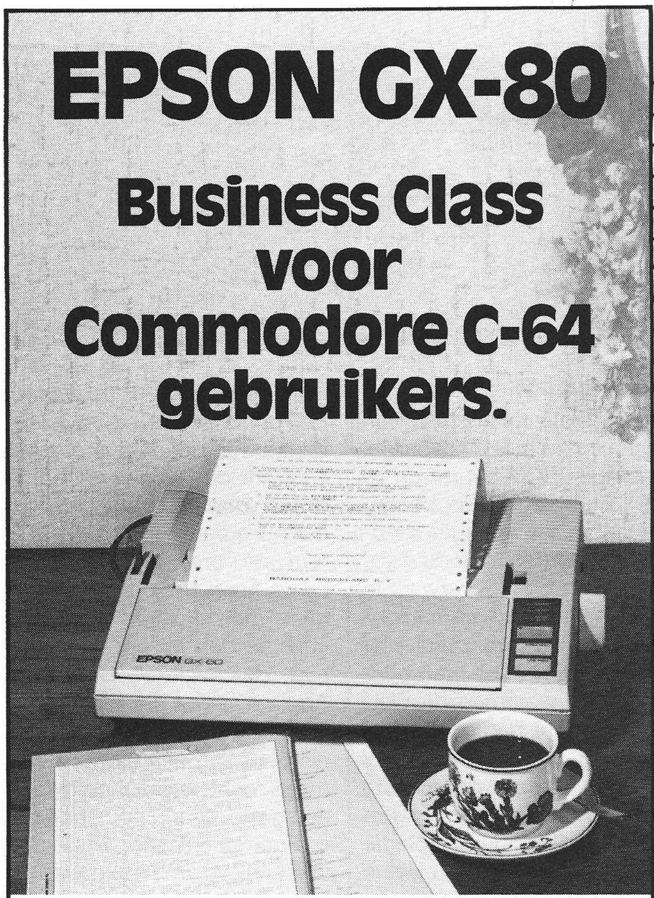
WE zijn wat voorzichtig met berichten over de Amiga, want je weet maar nooit wanneer CBM deze machine eindelijk in ons land zal gaan brengen. Maar in Amerika wordt het systeem nu verkocht en via de telex krijgen we heel wat enthousiaste berichten binnen van onze correspondenten.

Hoewel de machine nog niet door erg veel winkels gevoerd wordt, is hij te koop, inclusief systeemsoftware en ook beginnen de eerste Amiga toepassingen de winkelplanken te bereiken. Men is unaniem in het oordeel, dat de Amiga technisch een grote stap vooruit is. De paar systemen, die in Europa zijn, en men beweert dat ombouwen qua voltage etc. met weinig moeite te doen is, trekken veel belangstelling van software-huizen, al wacht men eigenlijk het liefst, tot ook de MS-DOS uitbreiding er is. CBM zelf geeft geen prijsindicatie, maar we schatten, dat voor rond de 6500 gulden voor een standaard systeem met 256 KB én kleurenmonitor de Amiga rond februari in ons land te koop zal zijn.

## Superscript

Al een tijdlang is het Easyscript tekstverwerkingspakket in de opruiming, voor minder dan vijftig gulden was het te koop en dat was voor de kenners al een teken, dat er iets nieuws aan kwam. Welnu Precision Software heeft de opvolger van Easyscript nu klaar en dat pakket heet **Superscript**. Een zeer uitgebreide tekstverwerker met alle funkties van Easyscript, maar iets overzichtelijker en gebruiksvriendelijker. En niet te vergeten met een hoop extra's. Er is een Nederlandse versie van in voorbereiding, waarover we nog nader zullen berichten.







## PCM Show

Wat de grote microshow van dit najaar had moeten worden, namelijk de PCM beurs van 10 tm 12 oktober, is qua bezoekersaantal een matige vertoning geworden. Beduidend minder dan de verwachte 50.000 bezoekers kwamen opdraven en dat lag zeker niet aan de immense promotiecampagne, waarmee de VNU dit evenement had rondgebazuind. Op de radio, in vrijwel alle VNU bladen, de meeste computerbladen, overal kwam men de PCM-Show reclame tegen, men heeft er een kleine vier ton aan promotie tegenaan gegooid. Volgende niet alleen gratis toegangskaarten, jongens, doe er voor dat geld maar een gratis treinkaartje bij!
Aan de vooravond van de beurs organiseerde men een grootse uitreiking van de Micro-Computer trofeeën, waar coryfeëen als Chriet Titulaer en Maurice de Hond bij optraden. Chriet weer met baard, zijn Sony reclame werd hem overigens door Ivo Niehe humoristisch onder de neus gewreven. Een prima avond, ook onze redaktie was het daar over eens en Eindeloos won verdiend de trofee voor het beste Commodore programma 1985.
Aan de organisatie van de beurs heeft het ook niet gelegen, men kwam uitstekend voor de dag en het was jammer, dat bv. IBM, Olivetti en Apple niet wat meer aandacht aan deze belangrijke gebeurtenis hebben besteed. Er is namelijk echt wel behoefte aan een grote, overkoepelende micro-computerbeurs. Maar dan wel één, waar de exposanten kunnen rekenen op 75 tot 100.000 bezoekers en de bezoekers van hun kant een breed aanbod kunnen verwachten. Maar met nogal wat computerbeurzen dit najaar leek het wel, of men een beetje beursmoe was. Overigens werden er toch wel goede zaken gedaan en liet bv. Commodore toch duidelijk zien, dat de C-128 in grote aantallen beschikbaar was, op de beurs en in de winkels.

## CES 1986 Las Vegas

**Nederlandse deelname aan Amerikaanse beurs voor consumenten elektronica.**

**De Nederlands-Amerikaanse Kamer van Koophandel, in samenwerking met de EVD bereidt een gezamenlijke deelname van Nederlandse bedrijven aan de CES (9-12 januari Las Vegas USA) voor. Dat is een ideale springplank om bv. computersoftware of randapparatuur onder de aandacht van de Amerikaanse pers en inkopers te brengen.**
**Er is de mogelijkheid van subsidie en belangstellenden kunnen contact opnemen met de heer van Dijk. ☎ 070-478234.**

## WINTER CONSUMER ELECTRONICS SHOW USA

## Gebruikersgroepen

De **VCGN** (070-971851) is de grootste Commodore gebruikersvereninging en zit iedere laatste zaterdag van de maand in het Lodewijk Makeblijde lyceum in Rijswijk.
Inlichtingen over de vele afdelingen kunt u krijgen via: Zeeland 01180-25746 West-Brabant 01640-51639 Midden-Brabant 01620-55634 Oost-Brabant 04120-42054 Zuid-Limburg 043-476935 Zuid-Holland 010-501023

**PET** Benelux heeft bijeenkomsten in Zeist (1e zaterdag), Nijmegen (2e zaterdag) en Haaksbergen (3e zaterdag). (Inl.05759-2211).

De **HCC** Commodore gebruikersgroep heet Compet, 03404-59599.

**HOT NEWS** van SCN uit Amsterdam (03450-16051) heeft haar grote bijeenkomst in de Meervaart op de 2e zaterdag van de maand en ook activiteiten in Zwolle (038-548459), Midden Nederland (03453-1375) en Deventer (05700-50758).

**WZW** is een van de clubs uit Belgie, met afdelingen in Antwerpen, Limburg, West-Vlaanderen inl. Bindstr. 19, te Berchem Antwerpen.

Ook in Gent is er een groep, te bereiken via 091-253937

In Arnhem zit ook een Commodore-club, waar men clubavonden op dinsdag iedere twee weken organiseert en ook cursussen geeft. Inl. Talingsingel 32 te Velp, tel. 085-647782.

In **Noordwijk** zit een C-64/128 gebruikersgroep, die iedere 1e donderdag een bijeenkomst heeft. Inlichtingen A vd Klaauw ☎ 01719-16720.

In het Belgische **Knokke-Heist** is ook een gebruikersclub actief. Men is daar heel ruimdenkend, de bijeenkomsten staan open voor gebruikers van alle merken home computers. Elke vrijdag vanaf 20.00 uur komen de computerenthousiasten uit de omgeving bijeen in Café Monty in Knokke. Op zaterdag komt men bij elkaar in de Broederschool in Duinbergen van 9.30 tot 12.00 uur. Het contactadres is: Computerclub Knokke Heist, p.a. Bob Dyserink, Kerkstraat 19, 8390 Knokke Heist, België.

De homecomputermarkt wordt overstroomd door zogeheten educatieve software. Ook voor de Commodore 64 is een er hele sortering in dat genre. Alle grote uitgevers zijn uit op een hap uit de koek. Vaak laat de kwaliteit te wensen over, men geeft op zich matige spelletjes de toevoeging educatief en dan moet het maar verkopen. Zo vinden we bijvoorbeeld de Nederlandse taal bij educatieve software wel een dwingende noodzaak, maar dat is niet de enige reden, waarom we toch wel blij waren met de pakketten, die Malmberg nu op de markt brengt. Onze onderwijsredacteur keek er naar.

# EDUCATIEVE SOFTWARE VAN MALMBERG

**Door Bob Munniksma**

**Het beoordelen van educatieve software is niet gemakkelijk, het is relatief eenvoudig om iets af te kraken, omdat het te eenvoudig of te moeilijk is. Maar aangezien de doelgroep zo enorm gevarieerd is qua leeftijd en interesse, kan wat voor de één snel verveelt, voor de ander een ongekende uitdaging betekenen. Wat grappig is voor de één, spreekt een andere leeftijd weer niet aan.**

Daarom is het zo goed mogelijk aangeven van de groep, waarvoor een bepaald programma bedoeld is, zo belangrijk. Malmberg zet dat duidelijk op de doos, maar toch vinden we de aanduidingen soms wel wat ruim. Voor de groep van 6 tot 10 bijvoorbeeld mag men in de praktijk best 7 en 8 jarigen lezen, als men zich even de commerciële motieven realiseert van de uitgever. Overigens blijft Malmberg heel duidelijk zijn image van kwaliteitsleverancier trouw, men heeft de zaak leuk verpakt, de documentatie is duidelijk en ook voor de kinderen te begrijpen, er zit een goede garantiekaart bij, men kan zich opgeven om later meer informatie van Malmberg te ontvangen, het is allemaal keurig verzorgd.

Dat het uiteindelijk gaat om vertaalde Amerikaanse programma's is een klein nadeel, de leermethodes etc. wijken daar toch wel iets af, maar dit wordt gecompenseerd door het feit, dat het gaat om produkten van de echte top-softwarehuizen zoals Spinnaker.




## SCHRIJVERTJE

Zoals de naam al doet vermoeden, is bij dit programma schrijven van verhalen het hoofddoel. Het programma is bedoeld voor kinderen van 6 tot 10 jaar. De jonge computervriend(in) is het "schrijvertje" al is het eigenlijk ook meer "illustrator", maar dat woord is te moeilijk als titel voor deze leeftijdsgroep. Het gaat namelijk om het maken van een prentenboek, en dat is met SCHRIJVERTJE een eenvoudige en vooral grappige bezigheid. Zoals de meeste pakketten uit de educatieve serie van Malmberg is ook SCHRIJVERTJE van origine Amerikaans. Spinnaker is voor velen een begrip op gebied van educatieve software. Het programma staat op disk, wat jammer is, omdat zo veel cassettegebruikers (en dat zijn de meeste C-64 bezitters!) er niet mee kunnen werken. Het blijkt echter onvermijdelijk, omdat anders het opslaan en

laden van de geschreven verhalen veel te omslachtig en zo gebruikersonvriendelijk wordt. De doelgroep (kinderen van 6 tot 10 jaar) zou er dan niet mee kunnen werken. Met de disk in de drive gaat alles als vanzelf.
De schijf zit samen met een gebruiksaanwijzing in de vorm van een leuk boekje, in een kleurige cassette. Malmberg houdt van klantenbinding, want aan de garantiekaart zit een antwoordkaart, bedoeld voor de marketing van Malmberg. In ruil voor wat gegevens krijgt de koper een gratis exemplaar van Malmbergs huiscomputer clubblad. De gebruiksaanwijzing zelf is duidelijk en geeft zonder veel omhaal de nodige programma-informatie. Eenmaal aan de slag met het programma, is de gebruiksaanwijzing vrijwel niet meer nodig, omdat op het scherm steeds alle nodige aanwijzingen staan. Met de opties op het scherm kunnen alle handelingen worden uitgevoerd. Er zit ook geen "dead end" in het programma. Steeds als het fout dreigt te gaan meldt de computer zich terug met een verzoek of een mededeling, zodat de gebruiker verder kan, zonder dat de boel vastloopt. Dit is natuurlijk een voorwaarde in een programma voor kinderen. Ook volwassenen blijken in staat een computer, vaak door omwetendheid, in een lock-up te krijgen; kinderen zijn helemaal niet gehinderd door enige technische computerkennis. Een programma voor kinderen moet daarom aan nog hogere fool-proof eisen voldoen. Schrijvertje voldoet aan die eisen.
Nadat er een plaatje is gemaakt, kan onze kleine auteur aan de slag met een "tekstverwerker" onder het plaatje, waarop een verhaal kan worden geschreven. Net als bij een echte tekstverwerker kan de tekst gewijzigd en aangepast worden, met enige beperkingen. Heel leuk allemaal. Daarna kan het plaatje met de tekst, als een enkele bladzijde of als eerste blad van een langer verhaal op disk worden opgeslagen. Steeds weer verschijnt de vraag aan de gebruiker, wat de volgende stap moet zijn: een algehele begeleiding van onze auteur in spé.
Het maken van het plaatje is een bijzonder uitgekiend programmadeel. Ook nu is er weer een menu voor de te kiezen handelingen. Er kunnen figuren worden uitgekozen uit een lange reeks. Is er een figuur gekozen, dan kan deze met de cursortoetsen op de gewenste plaats worden gezet. Kiezen voor groter of kleiner maken is ook mogelijk. Tijdens het ontwerpen kunnen steeds de achtergrondkleuren worden veranderd. E staat voor einde en dan kan het verhaal worden bedacht.

Alle opgeslagen bladzijden kunnen ook weer worden ingeladen en na elkaar worden bekeken. Heel verzord uitgevoerd zijn hierbij de meerstemmige stukjes muziek. Grote muzikale klasse.

### Vergelijking
Er zijn meer van dit soort programma's op de educatieve markt. Als ik Schrijvertje vergelijk met bijv. PlayWriters "Tales of me" van Woodbury, dan blijkt Schrijvertje veel smaakvoller en vooral pedagogisch verantwoorder. En Schrijvertje is natuurlijk geheel in het Nederlands!
Playwriter vraagt minder kreatieve inbreng van het kind, gaat alleen uit van een aantal feitelijkheden, die het programma dan verwerkt tot een min of meer aardig verhaal. Het kind kan dan het verhaal met kant en klare stickers illustreren en inbinden. Maar zoals gezegd, voor een andere leeftijdsgroep kan dat juist weer erg leuk zijn om Engels te leren. Voor bv. 10 tot 14 jarigen is dat weer een ander soort stimulans. Voor de jongere groep is Schrijvertje naar mijn mening te verkiezen.



*"Tales of me" kan de vergelijking met "Schrijvertje" niet echt doorstaan.*

## Fisher Price

Naast de software onder Malmbergs eigen label, is er nu ook een serie van vier educatieve spellen onder de naam Fisher-Price op de markt. **ALFABETSTAD** en **DENKRAAM**, beide voor de leeftijdsgroep van 4-8 jaar en **TUIMELTELLER** en **PIEKERSPOOR** voor de 7-12 jarigen komen eveneens van Spinnaker, ook al staat er een andere naam op de cassette. De naam Fisher-Price suggereert de onverwoestbaarheid van het kinderspeelgoed van oersterke kunststof, dat onder deze naam te koop is. En inderdaad, deze vier programma's kunnen wat mij betreft niet kapot, letterlijk en figuurlijk. Ze zijn niet alle vier even goed, maar steken toch ver uit boven wat er verder op de markt is in dit genre. Didactisch doordacht en grafisch zeer goed verzorgd en op cassette met turbolader. Naast het voordeel van een turbolader moet wel worden vermeld, dat bij het laden de kop van de recorder goed afgesteld moet zijn. Zelf moest ik eerst de kop bijstellen met een speciale afregelset, voordat ik naar de programma's kon kijken.

### Alfabetstad
Bij ALFABETSTAD moeten er letters worden opgezocht. De mascotte, een pinguin, moet de laddertjes op om een letter te halen. Die moet vervolgens in de juiste wachtende vrachtwagen worden gedeponeerd. Als er vier letters goed geladen zijn, vertrekken de wagens en komen leeg terug voor een volgende ronde. Zo wordt er een stad, Alfabetstad dus, gebouwd en geverfd. Het ziet er leuk uit. Alleen het beeld is wat saai en houdt de aandacht van een jong kind niet erg lang gevangen. Ook aan het muziekje had m.i. wat meer aandacht kunnen worden besteed.

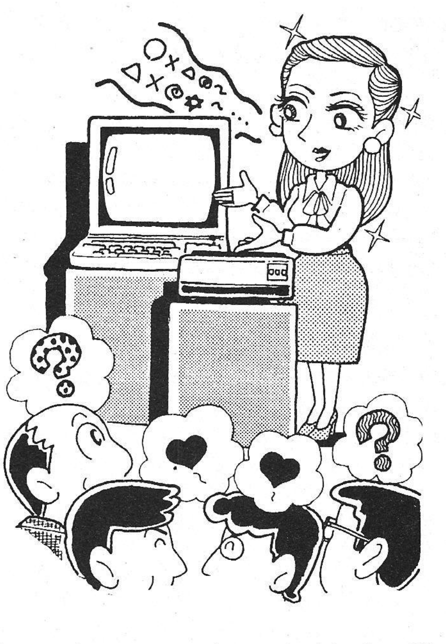
### Denkraam
Dit heeft weinig te maken met de vermogens van de onsterfelijke heer Ollie. B. Bommel, zoals die in de verhalen van Toonder door de professor worden beschreven. DENKRAAM zou je kunnen vergelijken met een memory spel. Achter verschillende raampjes in twee gebouwtjes zitten steeds twee dezelfde voorwerpen verborgen. Het gaat erom steeds twee gelijke te vinden, zoals dat bij memory ook het geval is.

## Tuimelteller

Het spel/leer programma TUIMELTELLER traint de elementaire bewerkingen van het rekenen. Optellen, aftrekken, vermenigvuldigen en delen worden met dit opwindende spel op een zeer aangename wijze geoefend. Het verveelt niet snel, doordat het spelverloop geheel door de speler(s) is in te stellen. Tegen elkaar of tegen de computer, met of zonder tijdslimiet. Ook kan de speler kiezen welke bewerking of combinatie van bewerkingen hij of zij wil. Dan gaat het erom, om door van getal tot getal te springen (via de gekozen bewerkingstekens), het getal, boven in beeld aangegeven, te bereiken. Hoe dan ook, de computer laat zich haast niet verslaan. Het blijft een onweerstaanbare uitdaging om het toch te proberen!

## Piekerspoor

Het programma PIEKERSPOOR laat de spelers(s) de logica van bepaalde situaties ontdekken om zo de gestelde problemen op te lossen en het spel te voltooien.



Voor al deze vier paketten geldt eigenlijk hetzelfde. Ze zijn zeker de moeite waard en van belang voor ouders die vinden dat hun kind ook eens met de computer mag kennismaken en daarvoor iets willen kopen. Veel ouders verkiezen de computerbeeldbuis boven de buis, die, meesal vlak na Sesamstraat, zijn geweld uitstrooit over jeugdig Nederland. Ik schaar mij in die rij en raad alle C-64 bezitters met jonge kinderen aan, het kijken naar de TV wat minder consumptief en meer actief en educatief te maken. Er valt nog heel wat te beleven en te leren zonder tv-series, ook voor ouders. Programma's als hierboven besproken kunnen daaraan een steentje bijdragen. ●

Met een geheugenuitbreiding tot 64 KB maak je van de C-16 zeker geen C-64, maar wel kom je qua geheugen net zo ver als de Plus/4.

C-16

# Geheugen uitbreiding

**Zoals bekend zit de C-16 niet erg ruim in zijn geheugencapaciteit. Met 16 KB loopt de gebruiker al gauw tegen de memory full boodschap aan en er zijn dus al wat C-16 bezitters op zoek naar een geheugenuitbreiding. Commodore zelf, die ooit heeft aangekondigd iets dergelijks te brengen, doet er ondertussen over de C-16 vrijwel het zwijgen toe, dus meot men elders gaan zoeken. Er zijn 16 KB en 64 KB uitbreidingen op de markt, voor prijzen van respect. ƒ 119,- voor de Stonechip Rampack (via o.a. Cafka) en ƒ 275,- voor de MCT64 via Infolist.**

De Rampack uitbreiding is vrij goedkoop, maar men komt dan toch nog op 28661 Byte vrij beschikbaar geheugen, dat zowel via Basic als in machinetaal-toepassingen te gebruiken is en dat is al behoorlijk wat vergeleken met bv. de Vic-20 of de zelfs de C-64. Het Rampack is een module en past achterin de C-16, dus geen sleutelwerk, maar de slot is dan natuurlijk wel bezet.

## Ramboard

Voor degenen, die denken dat met 64 KB RAM geheugen ook de compatibiliteit van de C-16 met de C-64 in een klap is verwezenlijkt, moeten we helaas vaststellen dat dat niet mogelijk is. De Basic-versie en vooral de I/O kernel van de C-16 wijkt behoorlijk af van die van de C-64, en die kloof valt niet of niet eenvoudig te overbruggen.

Wat wel mogelijk is, is het uitbreiden van het geheugen. Een zgn. RAMBOARD vergroot de beschikbare geheugenruimte van de C-16 met 16 KB of 64 KB, in dat laatste geval tot effectief 60.671 vrije bytes, waarmee de effectieve ruimte die van bv. de C-64 (38 KB) behoorlijk te boven gaat, men heeft in wezen bijna 70% meer ruimte en dat komt zeker bij database-toepassingen goed van pas.

Met de geinstalleerde RAM-board is de C-16 100% compatibel met alle bekende C-16 software en, dat is natuurlijk erg belangrijk, ook met de Plus/4 software.

De C-16 gedraagt zich dan als een Plus/4, indien tenminste niet gebruik gemaakt wordt van interfaces of andere hardware-eigenschappen van de Plus/4.

## Inbouw RAMBOARD

Het monteren van de kaart met extra geheugen in de machine is redelijk eenvoudig. Bij het monteren moet rekening gehouden worden met enige montage-ervaring, omdat tenslotte de zaak wel open moet en er voorzichtig met de pootjes van chips etc. moet worden omgesprongen. De operatie draait er om, dat de grootste videochip, 8360 of 7360, er uit wordt gehaald en er in het vrijkomende voetje een printplaat met chips komt en een nieuw verdiepinkje voor de videochip. Na controle en dichtmaken van de C-16 is er verder niets te zien, alleen de mededeling dat er dus ruim 60 KB beschikbaar is bij het opstarten.

We probeerden wat Plus-4 software, ook insteekmodules zoals LOGO en dat werkte perfect. Een bezwaar is de prijs, voor de aankoopprijs van een C-16 plus uitbreiding kan men beter direkt een Plus-4 kopen. Wie al een C-16 heeft, kan hiermee echter toch doorgroeien.

## C-16 software

Gelukkig komt er ook wat meer software voor de C-16, we signaleerden een relatief goedkoop Wordprocessing ën Database pakketje op cassette van Navajo software, dat door Audiogenic wordt gedistribueerd in Europa. Voor ƒ 79,- geen slechte zaak, daarmee wordt de C-16 weer ietsje interessanter. ●

In het vorig nummer gaven we al een hele berg detailinformatie over machinetaal en geheugenlokaties op de C-16. Daar bleken heel wat lezers, die een C-16 bezitten, heel gelukkig mee. Deze keer maken we de lijst met mogelijkheden van de TED chip af.

**Machinetaal Deel 2**

# TED's functies

## *Voor de C-16*



**Machinetaal op de C-16 is niet gemakkelijk, velen vonden het vorige artikel wel aan de pittige kant. Dat vinden wij ook, maar omdat we toch veel informatie over de C-16 kwijt willen, houden we het beknopt. Dat betekent helaas, dat de beginnende machinetaal-programmeur soms het spoor bijster raakt, maar we denken dat met wat studeren in algemene boeken over machinetaal dat wel valt in te halen. Dit artikel van Rob Vogelaar is meer bedoeld als een naslagartikel, goed om te bewaren als er details nodig zijn.**

We gaan weer verder met de funkties van de TED-chip, en daarbij waren we gebleven bij de interrupts.

**9 Raster Interrupts**

Dit is een moeilijk onderwerp, maar ik zal proberen het zo duidelijk mogelijk te maken. Het televisiebeeld bestaat uit 64000 pixels in een 320 bij 200 matrix, de C-16 stuurt de TV en de TV schrijft de signalen op het scherm pixel voor pixel en lijn voor lijn.

In de C-16 zit een register waaruit u kunt lezen waar op welke regel de TV het beeld opbouwt. Het zichtbare beeld bij de C-16 loopt van $00-$08. In lokatie $ff1d staat de huidige rasterpositie. Stel, dat we nu het bovenste beeld in graphics willen en het onderste in tekst, dan kunnen we **graphic 2/4** gebruiken, maar hoe werkt dat?

Er zit ook een register in de C-16's TED-chip waar we een waarde in kunnen zetten, dit is register $ff0b en als we hier nu de waarde $a3 in zetten dan zal er een interrupt plaatsvinden als het raster $a3 aan de beurt is om op het TV scherm getekend te worden. Dan zal ook bit 2 In $ff09 weer worden gezet. Er is ook nog een interrupt-enable register $ff0a, daarin kunt u bepaalde interrupts uit- of aanzetten. Normaal staat daar $a2. Als u bit 1 reset dan komen er geen interrupts meer en staat het systeem stil !!!!(niet doen).

Door dit soort grapjes kunnen we de C-64 capaciteiten met de C-16 toch weer beter benaderen. Wat denkt u bijv. van de nederlandse vlag als achtergrond of zelfs 6 verschillende kleuren. Of een klein gedeelte grafisch scherm wat weer geheugen oplevert of meer dan een character-set of multi- en hires door elkaar!

Alle raster technieken werken via de interrupt en die loopt via vector $0314 = $ce0e. Vanaf dit adres lopen alle handelingen die de processor elke 60ste van een seconde doet.

Er zit een routine bij voor het split screen effect, een routine voor de lengte van het sound, een om het toetsenbord te doorzoeken etc.

Als we nu alle onnodige routines schrappen, dan houden we een kleine routine over, net genoeg om het toetsenbord af te lezen en het knipperen van de cursor te zien. Dus geen graphic 2/4 meer en geen sound-of cassette-werk.

Deze routine houden we dan over :

```
ce48 lda $fb
ce4a pha
ce4b lda£$00
ce4d sta $fb
ce4f php
ce50 cli
ce51 jsr $db11
ce54 plp
ce55 pla
ce56 sta $fb
ce58 jmp $fcbe systeem interrupt - sprong naar einde (pla's en rti).
```

Nu gaan we zelf een programma schrijven wat gebruik maakt van raster technieken waarmee we de nederlandse vlag als achtergrond laten verschijnen.

We moeten dan op drie verschillende rasterplaatsen drie verschillende kleuren in het achtergrond kleurenregister plaatsen.

We willen onze interrupt routine op $2000 laten beginnen, dus schrijven we eerst een routine om de vector te verleggen.

```
a 1400 lda £$00
a 1402 sta $0314
a 1405 lda £$20
a 1407 sta $0315
a 140a brk
a 140b lda £$0e
a 140d sta $0314
a 1410 lda £$ce
a 1412 sta $0315
a 1415 brk
```

De tweede routine gebruiken we om alles weer normaal te maken met een normale interrupt.

We willen dus 3 kleuren, rood, wit en blauw.

Op de volgende rasters rood $00-$40, wit $40-$8a, blauw $8a-$00.

Deze waardes lijken niet allemaal precies logisch maar het werkt goed.

Hier volgt de routine (met tekst en uitleg).

```
a 2000 lda $ff09
a 2003 sta $ff09; zet ff09 met ff09 daardoor kan de interr. plaatsvinden.
a 2006 lda $ff1d; kijk naar het raster register.
a 2009 cmp £$03; ligt het tussen $00-$03
a 200b bcs $201a; nee dan verder
a 200d lda £$52; ja, dan achtergrondkleur rood
a 200f sta $ff15; zet $52=rood in achtergrond register
```

a 2012 lda £$40;de volgende interrupt moet plaats hebben als het raster op $40 staat
a 2014 sta $ff0b;
a 2017 jmp $fcbe; einde van de interrupt
a 201a cmp £$43; ligt het raster tusen $40-$43
a 201c bcs $202b; nee dan verder
a 201e lda £$f1; ja, dan achtergrondkleur wit
a 2020 sta $ff15; zet $f1=wit in achtergrond register
a 2023 lda £$8a; volgende interruptie als raster op $8a is
a 2025 sta $ff0b
a 2028 jmp $fcbe; einde van deze interruptie
a 202b cmp £$8d; ligt het raster tussen $8a-$8d
a 202d bcs $2039;nee geen kleurverandering dus naar eind routine
a 202f lda £$46/ja dan achtergrondkleur blauw
a 2031 sta $ff15; zet $46=blauw in achtergrond register
a 2034 lda £$00;de volgende interruptie heeft plaats als raster $00 is.
a 2036 sta $ff0b
a 2039 lda $fb;hier volgt de ROM routine die nodig is voor het toetsenbord
a 203b pha; en de cursor
a 203c lda £$00
a 203e sta $fb
a 2040 php
a 2041 cli
a 2042 jsr$db11
a 2045 plp
a 2046 pla
a 2047 sta $fb
a 2049 jmp $fcbe

Het laatste gedeelte vanaf $2039 is het toetsenbord gedeelte waarover al eerder werd gesproken.

Er zijn nu wat dingen die veranderd kunnen worden:

➤ De kleuren op lokaties $200e, $201f, $2030.

➤ De raster posities, die wel van klein naar groot gekozen moeten worden: op lokaties $2013, $2024, $2035 achtereenvolgens de posities voor: de eerste, tweede, en laatste kleurovergang. Op de volgende lokaties moeten dezelfde waardes gezet worden maar dan met drie opgehoogd. Dit moet omdat er tussen de tijd dat er een interrupt plaats heeft en de processor bij de cmp op code is er nogal wat instrukties voorbij gaan en het raster is dan alweer wat verder. De lokaties voor deze waarden staan op $200a, $201B, $202c.

Hier volgt welke lokatie bij welke hoort voor het met 3 ophogen $2013-$201b, $2024-$202c, $2035-$200a.

➤: verander beide jmp$fcbe naar jmp$2029. Dit heeft tot gevolg dat de toets routine vaker wordt doorlopen, waardoor de cursor veel sneller beweegt.

### ⑩ De 8 Volume levels

De C-16 kan ook geluid produceren, weliswaar niet zo mooi als bij zijn broertje de C-64, maar toch nog twee kanalen. Voor geluid moeten we in de volgende registers kijken $ff11, $ff0e, $ff0f, $ff10, $ff12.

Voor het volume kijken we in het register $ff11 en wel de bits 3- 0. De combinatie 0000 is uit, 0001 is zacht en 1000 is het hardst.

Voor het kiezen van bepaalde stemmen moeten we ook in $ff11 kijken.

bit 4 = voice 1    1 = aan/0 = uit
bit 5 = voice 2    1 = aan/0 = uit
bit 6 = voice 2    1 = white noise
0 = no noise

voice 1 = $0001
voice 2 = $0010
voice 1 + 2 = $0011
voice 2 = noise = $0100
voice 2 = noise + voice 1=$0101

De frequentiewaarden voor voice 1 zijn: $ff12, $ff0e
$ff12 bit 1-0 = hi value
$ff0e = lo value

### Muziek maken

Hier volgt een kleine routine als voorbeeld hoe muziek uit de C-16 te krijgen valt m.b.v. machinetaal:

```
a 2000 lda £$18
a 2002 sta $ff11;        zet volume en voice 1
a 2005 lda £$00
a 2007 sta $ff0e;        clear lo. val.
a 200a lda $ff12
a 200d and £$fd
a 200f sta $ff12;        clear hi. val.
a 2012 inx
a 2013 bne $2012;        delay loop
a 2015 lda $ff0e
a 2018 clc
a 2019 adc £$01
a 201b sta $ff0e;        verhoog frequentie
a 201e bne $2012;  lo. val. overschreden?
a 2020 lda $ff12;        ja, verhoog lo. val.
a 2023 clc
a 2024 adc £$01
a 2026 sta $ff12;verhoog hi. val.
                   a 2029 and £$03
a 202b bne $2012;  hi. val. overschreden?
a 202d lda £$c4;         ja, clear de registers
a 202f sta $ff12;        clear hi. val.
a 2032 lda £$08
a 2034 sta $ff11;        set voice 1 off.
a 2037 brk;              back to tedmon
```

Als u dit runt (g't) dan hoort u een oplopende toon, die vergeleken kan worden met het Basic resultaat van :

```
10 vol 8
20 for x=0 to 1000
30 sound 1, x, 1
40 next x
```

Ten eerste doet Basic er veel langer over om alle (step=1) hoogten te laten horen en ten tweede zijn er tikjes doorheen te horen, omdat het geluid via interrupt gestuurd wordt.
Deze vergelijking valt dus duidelijk uit in het voordeel van het machinetaal programmaatje.

### ⑬ Screen blanking

Bij screen-blanking krijgt het hele scherm de randkleur. Dit wordt wel gebruikt om het scherm uit te zetten en terwijl het uit is wordt het scherm beschreven en daarna weer aangezet.
Het lijkt dan of het scherm in een fractie van een seconde gevuld word. Om het maak-proces niet in beeld te brengen wordt het scherm even geblankeld, de tekst blijft wel op het scherm, maar niet zichtbaar. De commando's hiervoor zitten in het control-register.
$ff06  bit 4  1 = tekst/0 = blank
Kleuren worden als volgt in getallen voorgesteld.

**A. Hoofdkleuren**, deze staan in de bits 3-0 en wel als volgt:

| | |
|---|---|
| 0000 = zwart | 0001 = wit |
| 0010 = rood | 0011 = cyaan |
| 0100 = paars | 0101 = groen |
| 0110 = blauw | 0111 = geel |
| 1000 = oranje | 1001 = bruin |
| 1010 = geelgroen | 1011 = rose |
| 1100 = blauwgroen | 1101 = lichtblauw |
| 1110 = donkerblauw | 1111 = lichtgroen |

**B. Helderheden**, dat zijn de bits 6-4 en wel als volgt:

| | |
|---|---|
| 000 = light | 001 = luninantie 1 |
| 010 = luninantie 2 | 011 = luninantie 3 |
| 100 = luninantie 4 | 101 = luninantie 5 |
| 110 = luninantie 6 | 111 = dark |

Een voorbeeldje 0,0011,0001 =$31 = wit met een iets donkere tint = grijs, in lokatie $053b wordt de voorgrondkleur grijs.
Bit 7 is bij de achtergrond, rand en multicolor registers altijd 1, bij de voorgrondkleur wordt dat bit gebruikt om aan te geven of het character wel of niet moet knipperen 1=knipper, 0=geen knipper.
Als van de $31 op lokatie $053b $b1 gemaakt wordt, begint de volledige tekst te knipperen. Met bit 7 van het kleurenscherm knipperen bepaalde letters, bv. bit 7 van lokatie $0800 verzorgt het knipperen van de letter in de linkerbovenhoek van het scherm.

**Tot zover deze ontdekkingsreis in de machinetaal diepten van de C-16 en de TED chip daarin.** ●

# Basic Min

Een rubriek van Nico Baaijens m

Voor deze Commodore-Info zijn niet minder dan 39 mini- programmaatjes binnen gekomen. Over gebrek aan creativiteit en zelfwerkzaamheid onder de lezers dus geen klachten. Ook blijken onze Belgische vrienden elke keer weer van de partij te willen zijn in deze rubriek en ook dat stemt tot tevredenheid.

**Ondanks het grote aantal inzendingen, kan ik deze keer helaas geen Tien met een Griffel uitreiken. Zoals in de vorige rubriek al werd benadrukt, moet dit een echte onderscheiding blijven, zodat ik hem pas weggeef als ik na het intypen van zo'n inzending en het geven van RUN juichend uit mijn stoel opveer en kan uitroepen: "Dat is 'm: de Tien met de Griffel!"**
**Even voor de statistiek: de T.m.d.G is tot nu toe één keer verleend. Eén keer was er een Tien min met een Griffel.**

We beginnen met enkele inzendingen van Richard Vierstra (15) uit Oterleek (dat schijnt in Noord-Holland te liggen). Hij zond een heel pak miniatuurtjes in, waarvan ik er drie uitkoos. Het meest opvallend was zijn grafische effect: Golvende Zee, dat bruikbaar kan zijn als effect in een game.

```
10 PRINT "<SHIFT CLR/HOME>":FOR A=1864TO2023
20 IFA/2=INT(A/2)THENPOKE A,77:GOTO40
30 POKE A,78
40 POKE A+54272,1:NEXT A:POKE 53281,6
50 POKE 53280,6:PRINT TAB(13)"<CRSR NEER CYAAN
RVS ON> GOLVENDE ZEE <RVS OFF>"
60 FOR D=1TO10:FOR B=0TO15:POKE 53270,B
70 GOSUB100:NEXT B:FOR C=15TO0STEP-1
80 POKE 53270,C:GOSUB100:NEXT C
90 NEXT D:POKE 53270,200:END
100 FOR I=1TO50:NEXT I:RETURN
```

```
0 REM 64 KLUTS
1 FOR A=0TO255:POKE 53270,A:NEXT A
```

```
0 REM HET BLOED KRUIPT!
1 PRINT "<SHIFT CLR/HOME>":POKE 53281,0:POKE 53280,0
2 POKE 54296,15:POKE 54277,7:POKE 54278,7
3 FOR A=55296TO55312:POKE A,3:NEXT A
4 FOR B=1024TO1040:READC:POKE B,C
5 POKE 54276,129:IFB=1024THEN7
6 POKE 54272+(B-1),2
7 POKE 54273,68:POKE 54272,149
8 POKE 54276,128:FOR I=1TO250:NEXT I
9 NEXT B:POKE 55312,2
10 DATA8,5,20,32,2,12,15,5,4,32,11, 18, 21, 9, 16, 20, 33
```

## Commodore op hol

Opvallend waren ook de miniatuurtjes en tips, die Erik van der Sandt uit Zelhem instuurde. Van hem leerde ik hoe ik het C-64 scherm naar omlaag kan laten scrollen met: SYS 59761 en omhoog met SYS 59765. Even simpel is het om een tekenprogramma te maken met: POKE 788,81. Helemaal dol wordt het met zijn onderstaande miniatuurtje, waarmee de C64 totaal de kluts kwijt raakt en op hol slaat. Heeft dit iets te maken met bugs in de Basic-interpreter?

```
10 FOR I=1TO1000000:PRINT "<CTRL-2>"CHR$(PEEK(I
   ));:NEXT
```

Een merkwaardige manier om het C64-scherm vol te schrijven met vierkante blokjes stuurde Remon Witteveen uit Heerhugowaard in:

```
10 PRINT "<SHIFT CLR/HOME>":POKE 53280,0
20 POKE 53281,0
30 FOR R=1024TO1983
40 FOR W=64TO250
50 POKE R,W
60 NEXT :NEXT
```

Iets in die geest kwam van Ralph Görtzen uit Geleen. Na RUN wordt op de Shift Lock toets gedrukt en wat vervolgens wordt ingetypt, verschijnt in Hi-Res. Met de getallen achter de komma's kan worden geëxperimenteerd om fraaie kleurcombinaties te krijgen.

```
1 V=53280:POKE V,0:POKE V+1,0:POKE V+2,0:POKE V+3,1
  :POKE 646,6
2 POKE 53265,PEEK(53265)OR64
```

Het C64-beeldscherm en zijn rariteiten blijven een bron van inspiratie voor Basic-miniaturisten. Niet helemaal origineel, maar toch de moeite van het intypen waard is het spiegelschrift van Hans Duivenvoorden uit Leiden.

```
100 DIMK(1000)
110 FOR T=0TO999
120 K(T)=PEEK(1024+T)
130 POKE 1024+T,160
140 NEXT T
150 FOR T=999TO0STEP-1
160 POKE 1024+T,K(ABS(T-999))
170 NEXT T
```

Als subroutine in een groter Basic-programma is de alternatieve manier om op een aandachttrekkende wijze de C64-gebruiker een tekstregel te laten lezen. Het miniatuurtje is van J.J. van der Hart uit Utrecht.

# iatuurtjes

orte tot zeer korte programma's.

```
10 A$="INFO OVER COMMODORE,LEES COMMODORE INFO"
20 V=54272:FOR T=V TO V+23:POKE T,0:NEXT :POKE T,15:POKE V+5,3
30 FOR T=1TOLEN(A$):PRINT MID$(A$,T,1);:FOR X=1TO 50:NEXT
40 POKE V+4,32:POKE V,177:POKE V+1,25:POKE V+4,33:NEXT
50 FOR T=1TO1000:NEXT :FOR T=LEN(A$)TO1STEP-1:PRINT CHR$(20);:NEXT
60 RUN
```

Gebruikers van Simons Basic worden ook steeds talrijker en geen wonder ook want daarmee wordt veel meer mogelijk, ook in de minatuurtjes. Onderstaand miniatuurtje loopt alleen onder Simons Basic en is van Vincent Houtsma uit IJlst.

```
10 HIRES0,1
20 FOR A=1TO320STEP5:LINE150,100,A,200,1:LINE 150,100,A,1,1:NEXT
30 FOR A=1TO200STEP5:LINE150,100,320,A,1:LINE 150,100,1,A,1:NEXT
40 GOTO 40
```

Nog een miniatuurtje voor Simons Basic van Richard Phielix uit
Zeist:

```
10 HIRES1,0
20 X=INT(200*RND(0))
30 Y=INT(200*RND(0))
40 LINE1,X,320,Y,1
50 GOTO20
```

## Artistiek

Met het miniatuurtje van J. Smeets uit Zevenaar schrijft de C-64 artistieke figuren op het scherm. Uiteraard is daarin de RND-functie actief. Het programma levert een aardig kijkspel op.

```
1 X=0:Y=0
10 POKE 53280,0:POKE 53281,0:PRINT "<SHIFT CLR/HOME>"
20 A=INT(RND(TI)*4)+1
30 IFA=1THENX=X+1:IFX>40THENX=40
40 IFA=2THENX=X-1:IFX<0THENX=0
50 IFA=3THENY=Y+1:IFY>24THENY=24
60 IFA=4THENY=Y-1:IFY<0THENY=0
70 POKE 1024+X+Y*40,65:POKE 55296+X+Y*40,2:IFJ=0THENJ=1:GOTO30
80 J=0:GOTO20
```

Kleine spelletjes blijven natuurlijk ook welkom. Een aardig voorbeeld van een mini-game geeft Jankoen de Haan uit Arnhem. De grote uitdaging is om met de s- en a-toetsen het ruimteschip te besturen en door de sterren heen te laveren.

```
5 Z=1224
10 Q=INT(RND(1)*38):PRINT TAB(Q)"*"
20 IFPEEK(Z+X)=42THENEND
30 GETA$:IFA$="A"THENX=X-1
40 IFA$="S"THENX=X+1
50 POKE Z+X,160
60 POKE 55496+X,7
70 GOTO10
```

## C-16

"Omdat de miniatuurtjesrubriek toch vol moet", schrijft Ivar Iding uit Apeldoorn, "heb ik ook maar wat geschreven voor de C-16". Het is een scroll-routine, die van pas kan komen.

```
10 COLOR4,1:COLOR0,1:COLOR1,2
20 DO UNTIL A$=" "
30 GETA$
40 SCNCLR:PRINT SPC(15)"*SCROLLEN*"
50 GETA$
60 FOR X=1TO23:PRINT CHR$(27)+CHR$(87)
70 GETA$
80 FOR Y=1TO100:NEXT :NEXT
90 GETA$
100 FOR X=1TO23:FOR Y=1TO100:NEXT :PRINT
110 GETA$
120 NEXT :LOOP
```




**Al met al weer een substantiële Basic-miniatuurtjes-rubriek. Ik heb helaas niet alle inzenders kunnen honoreren met plaatsing in dit veelgelezen blad. Er blijven nog wat inzendingen op de plank liggen, maar ik zie toch weer halsreikend uit naar de nieuwe inzendingen voor de volgende Commodore Info. Zal er een Tien met de Griffel bij zijn? Ik ben benieuwd.**

**Nico Baaijens**

**In het vorige nummer van Commodore Info gaven we onder de titel "C-128 : opvoerset hard nodig" een beschouwing over de nieuwste telg van Commodore. We hebben voor dat artikel de C-128 op een aantal gebieden onder de loep genomen en voor zover we daartoe in de gelegenheid waren, alle goede en minder aantrekkelijke kanten van die machine weergegeven. Commodore zelf vroeg ons echter, toch nog eens naar de machine te komen kijken, omdat ze vonden, dat bepaalde dingen wat nadere aandacht verdienden. Met name zou de 128 in de standaard configuratie veel beter werken dan in de halve 64/128 constellatie met de 1541, die we de vorige keer bespraken.**

# Opnieuw de C-128



**De door ons uitgevoerde tests op de C-128 gaven aanleiding om bij een aantal eigenschappen van het apparaat nogal wat kritische kanttekeningen te plaatsen. Zo moesten we constateren dat met name de verwerkingssnelheid zowel in Basic als in machinetaal in de 128-mode aanmerkelijk trager was, dan we gewend zijn van z'n voorloper, de C-64. Daarnaast gaven we ook de nodige kritiek op het verschil in monitor- aansturingen voor 80- en 40-koloms display en de implementatie van het CP/M 3.0 operating systeem op de C-128.**

Omdat we in de initiële tests een konsekwent gebruik gemaakt hebben van de (bestaande) 1541 diskdrive, heeft Commodore Nederland ons nu een paar exemplaren van de nieuwe, voor de C-128 ontworpen, 1570/71 diskdrives ter beschikking gesteld plus een 128 monitor, waardoor we de C-128 nu in z'n standaard configuratie hebben kunnen gebruiken. Of dit enige verbetering brengt in het totaalbeeld van de Commodore PC 128 kunt u hieronder lezen.

## Het C-128 concept

Sinds de aankondiging van de C-128 in 1984 zijn er veel speculaties gemaakt op de mogelijkheden die het nieuwe wonder van Commodore ons zou gaan bieden. Naarmate het wachten langer duurde, kwamen enerzijds de verhalen los van grote blunders bij het ontwerp, anderzijds werden de speculaties over het apparaat nog fantastischer. Nu de machine (eindelijk) in Europa is uitgebracht kunnen we de verwachtingen toetsen aan de werkelijkheid.
Om de capaciteiten van de C-128 goed te kunnen beoordelen moeten we uitgaan van de doelgroep waarvoor deze computer is ontwikkeld. De filosofie van het ontwerp richt zich hierbij op de gevorderde 64-gebruikers. In opzet moet je de C-128 bekijken als de grote en volwassen broer van de C-64, of -zo je wilt- de grootvader van de VIC-20 en de oom van de C-16 en Plus/4. In verhouding tot de PC10/20 en andere IBM-clones is de C-128 een verre neef, die wel de klok heeft horen luiden, maar de klepel hangt in de PC. Los van familierelaties komt het erop neer, dat de C-128 niet is gemaakt voor langdurig commercieel gebruik. Veel meer is gemikt op de voortgang in computerland om door te groeien vanaf VIC en C-64. Daarom is bij het ontwerp zoveel aandacht gegeven aan de compatibiliteit (het verenigbaar zijn) met de C-64 standaard.
Om ook de groeimogelijkheden zo groot mogelijk te maken, is besloten de C-128 te voorzien van een extra processor, waardoor het CP/M operating systeem binnen de grenzen van de huiscomputer werd gebracht. Als laatste moest ook een uitgebreide en veelzijdige Basic-interpreter op het systeem worden opgenomen terwijl ook het beschikbare RAM geheugen moest worden vergroot.
Het resultaat van dit concept is nu dat we drie computers in één kast hebben gekregen waardoor de C-128 een tussenstap kan vormen tussen huiscomputers zoals de 64 en MSX en de grotere systemen waarbij de PC-achtigen op de eerste plaats komen.

## Drie computers

De drie computers waarin de C-128 kan worden veranderd verschillen onderling nogal wat van elkaar. Als eerste de **64-mode**, waarin de C-128 opereert als een C-64. Dit gaat zelfs zover, dat de op zich uiterst handige extra toetsen op het toetsenbord, zoals b.v. de numerieke 'pad' niet werken. Van het 128KByte geheugen wordt ook slechts 64K benut. Het scherm blijft hierbij 40-kolommen en de verwerkingssnelheid gaat uit van de C-64 1MHz klok- frekwentie. Daardoor wordt bereikt dat **alle** bestaande C-64 software op de 128 kan worden gerund.
De normale **C-128 mode** wordt verdeeld in twee segmenten. De 128/40koloms en 128/80 koloms verwerking. Bij het aanzetten van het apparaat kan door middel van een toets worden aangegeven, in welke display-mode (40 of 80 kolommen) moet worden gestart. De 128 zorgt dan automatisch dat de juiste routines de monitor aansturen. In 128-mode beschikken we over 128KByte RAM (voor Basic 122365 bytes free) waarbij globaal 64K bestemd is voor het programma en een 64K bank voor de Basic variabelen. Het blijft een 8-bits machine, waardoor rekening moet worden gehouden met het zogenoemde **Bankswitching**. Door dit grote geheugen wordt de gebruiker in staat gesteld min of meer professioneel te kunnen werken met een tekstverwerker (veel tekst tegelijk in het geheugen) een database of een spreadsheet. Zelfs geïntegreerde software (JANE) behoort dan tot de mogelijkheden. Het RAM- geheugen is zelfs uit te breiden tot 512Kbyte en dat maakt de 128 op dat gebied vergelijkbaar met bv. de IBM PC en de Macintosh.

## Hogere kloksnelheid

De 128-mode maakt gebruik van een zeer uitgebreide Basic, versie 7.0, die beschikt over meer dan 140 commando's, afgeleid van de C- 16 Basic 3.5 versie en aangevuld met de inmiddels voor de C-64 beschikbare utility-opdrachten voor Hires, geluid en machinetaal-assembly. Vervelende PEEKes en POKEs behoren hiermee tot het

verleden. De kloksnelheid van de 8502 processor is 2Mhz, het dubbele van de 6510 uit de C-64!
De derde mogelijkheid is de **CP/M-mode.** Deze berust op de werking met het 'Control Program voor Microprocessors' operating systeem. Digital Research CP/M 3.0 wordt op disk meegeleverd. Nadat de CP/M van disk is ingelezen hebben we de beschikking over 128KByte geheugen en het volledige toetsenbord, inclusief alle speciale toetsen. Deze mode verruimt de blik naar alle beschikbare CP/M software, die indertijd voor commercieel gebruik is ontwikkeld. Bekende programma's zoals WORDSTAR, PERFECT CALC en DBASE II kunnen in deze mode op de 128 worden gebruikt.

## Randapparaten

In het gebruik en de aansturing van de nodige randapparatuur zoals datasette, printer en diskdrive zitten nogal wat mogelijkheden. De cassetterecorder kan worden toegepast in 64- en 128-mode. Als **diskdrives** komen de 1541 en de nieuwe 1570 en 1571 in aanmerking. Alle seriële Commodore printers kunnen worden gebruikt, terwijl met een goede interface ook de meeste parallel Centronics regeldrukkers kunnen worden aangesloten. De schermuitvoer levert hier en daar wat problemen op, maar die zijn in principe allemaal op te lossen door gebruik van de juiste kabel, uitgang en monitor. Onderscheid moet worden gemaakt tussen 40- en 80-koloms monitoren, of een hardware combinatie van beide. Bij het gebruik van 40 kolommen wordt de VDU-chip van de C-64 benut. Dit betekent, dat daardoor in **High-resolution** kan worden gewerkt, maar ook dat de kloksnelheid van de totale verwerking wordt teruggebracht tot 1MHz. Al met al komt het erop neer, dat de voor de C-64 gebruikte monitor meestal niet geschikt is voor de 80-koloms display van de CP/M en 128/80-mode. Dit is alleen mogelijk als de monitor (kleur en monochroom) is uitgerust voor beide soorten invoer.

## Snelheid

De verwerkingssnelheid van Basic hangt direkt samen met de gebruikte monitor. Wat de 64-mode betreft blijkt de C-128 exact even lang te doen over het runnen van een programma als de oude C-64 zelf. Dit is anders in de C-128/40-koloms mode. Hoewel dat de enige manier is om met de C-128 gebruik te maken van Hires en sprites en derg. wordt hier de C-64 Videochip voor 40 kolommen gebruikt, waardoor de 128 klokfrekwentie wordt teruggebracht tot die van de videochip, dus 1MHz. Daar komt bij, dat in 128-mode gebruik wordt gemaakt van twee 64K-geheugenbanken, waardoor tijdens het runnen steeds moet worden geschakeld van de ene bank naar de andere. En dat kost zoveel tijd, dat hetzelfde programma er in 128-mode langer over doet dan de C-64, iets wat we ook in ons vorig artikel constateerden.
Gebruiken we daarentegen de 128/80-koloms mode, met dito monitor, dan komen we heel andere uitkomsten tegen voor dezelfde programma's. Daarbij wordt namelijk gebruik gemaakt van de 80- koloms videochip en de eigen 2MHz klok. En dan blijkt de 128 ineens veel sneller dan de C-64.
Om dezelfde snelheden te bereiken in 128/40-mode beschikt Basic 7.0 over het commando **FAST** waardoor de video (lees monitor) wordt uitgeschakeld, en de 2MHz klok wordt gebruikt om de software te verwerken. Dit is hetzelfde effect als we kennen bij het laden van een cassetteprogramma op de C-64. Nadat de opdracht **SLOW** wordt gegeven is het scherm weer terug en werkt de 128 weer op 1MHz.
De reden van deze klok-truuk is mij niet duidelijk, wel is het natuurlijk zo, dat er al zoveel in de C-128 zit, dat er waarschijnlijk niet nóg een videoprocessor bij kon voor de 128/40 koloms mode. Ter verduidelijking een paar uitkomsten van standaard testen (in Basic) op de diverse modes in vergelijking met de C-64.

## Diskdrive

De conclusies van de eerdere testen die we met de C-128 hebben uitgevoerd, waren wat disk-verwerking betreft niet gebaseerd op de volledige capaciteit van de C-128, omdat we uitsluitend de beschikking hadden over een 1541 drive. Hoewel het gebruik van de C-128 in combinatie met de 1541 probleemloos is, werkt de data-over-

## RUNTIME -test (benchmarks)

| PROG | C-64 | 128/64 | 128/40 | 128/80(FAST) |
|---|---|---|---|---|
| Test 1 | 840 | 842 | 1265 | 598 jiffies |
| Test 2 | 108 | 106 | 213 | 61 " |
| Test 3 | 11615 | 11618 | 12793 | 6084 " |





Externe gegevensopslag is met een cassetterecorder altijd een beperkte en langzame zaak. Wilt u meer en sneller, zelfs veel meer dan een diskdrive, dan is er nu de Stardrive, een magnetisch medium met behoorlijke kwaliteiten.

# Stardrive: 1000 KB op cassette.

**De diskdrive is voor velen een fel begeerd, maar te prijzig bezit. Een Commodore 1541 (model 1985) kost in de meeste winkels rond de ƒ 800,- en dat kan de hobbybeurs vaak niet opbrengen. Het is dan ook niet verwonderlijk dat de aanzienlijk goedkopere datarecorders uitgerust met een snellader of turbo tool een grote populariteit genieten. Maar het kan nog anders, een nieuwe ster aan het data-opslag firmament kan de "STARDRIVE" van het Amerikaanse bedrijf Windsor Electronics worden.**

De STARDRIVE Turbo Tape Transport model FT1500 datarecorder gebruikt een qua uiterlijk standaard Philips audiocassette, alleen is die gevuld met een eindeloze tapelus van zeer hoge kwaliteit. Die cassette draait ook veel sneller dan normaal en kan beschreven worden met een hoge schrijfsnelheid. De voor cassetteband ongebruikelijke lees/schrijfsnelheid van 10 inches per seconde geeft een opslag capaciteit van 125 KB per seconde en voor snelspoelen kan het nog sneller! En daar kan menige diskdrive niet aan tippen. Signaaloptekening kan zowel via FM als NFM-codering.
Een bekend nadeel van dataregistratie op cassettebandjes is de wat hogere kans op fouten door drop outs en "informatieslijtage". Kortom cassettes zijn een minder betrouwbaar opslagmedium dan floppy disks. Windsor Electronics geeft voor haar produkt echter een hoge mate van betrouwbaarheid op: een levensverwachting van maar liefst 6000 gebruikscycli en een kans op harde fouten van 1 op 1.000.000.000.000. Zeker voor amateurgebruik geen getallen om je echt zorgen over te maken.
De C-64/128 versie zal volgens de fabrikant binnenkort tegen een scherp concurrerende prijs op de Nederlandse markt verkrijgbaar zijn, men spreekt over 390 gulden voor het 1 MB model, terwijl men een 4 MB versie voor de PC op het oog heeft.

Een STARDRIVE biedt veel "diskfaciliteiten" voor een verrassend lage prijs. De FT1500 kan gemakkelijk een diskdrive vervangen of als tweede drive functioneren. ●

## Opslag capaciteit Stardrive

De opslagcapaciteit is afhankelijk van de bandlengte van de digitale cassette.

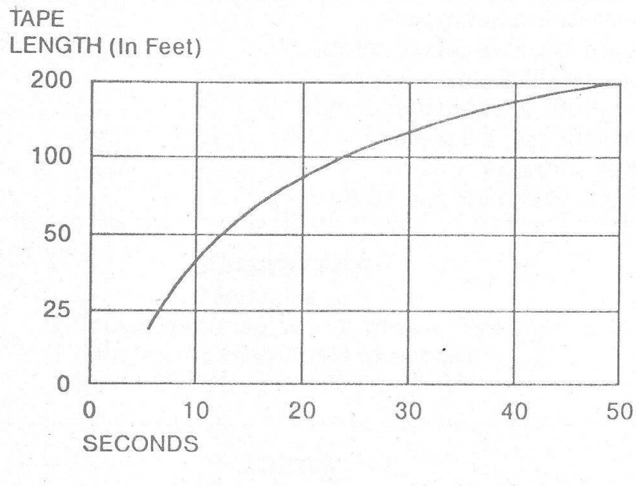
| Bandlengte: | Ongeformatteerde opslagcapaciteit: | Access/ Toegangstijd |
|---|---|---|
| 25 ft | 125 KB | gem. 6 sec. |
| 50 ft | 250 KB | gem. 12 sec. |
| 100 ft | 500 KB | gem. 24 sec. |
| 250 ft | 1000 KB | gem. 48 sec. |

(NB: ft=amerikaanse/engelse feetmaat)

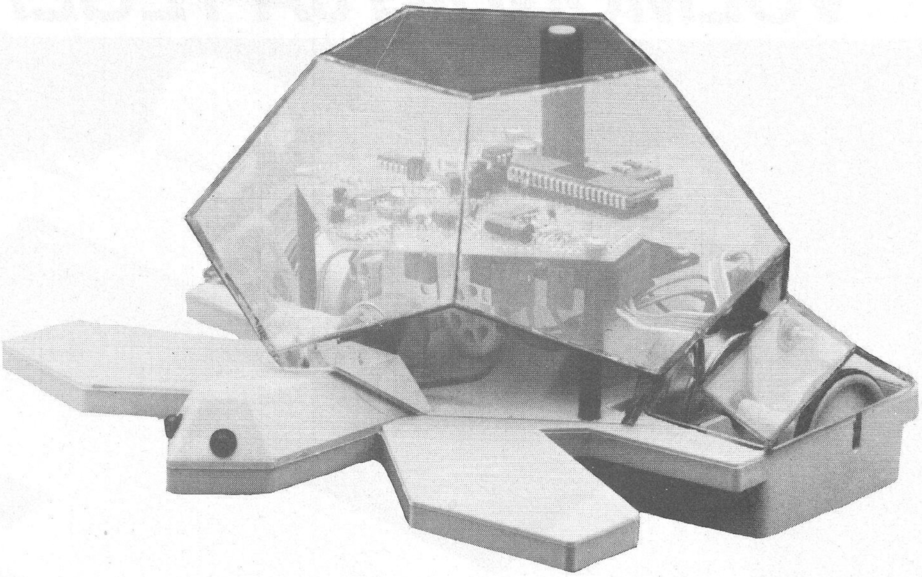
De maximum accesstijden bedragen het dubbele van de opgegeven waarden.

Overige gegevens:
Tape : digitale kwaliteitsband.
Snelspoelen 30 inch per seconde.
Onderhoud iedere 40 gebruiksuren met isopropyl alkohol en Wattenstokje.
Daisychain poort op de recorder.
Aansluiting C-64 op de cassette poort.
Mogelijkheden C-64: Load, Save, Copy (tussen STARDRIVES en disk) en file-overdracht naar de printer.

## Average Access Time Chart



# IN GESPREK MET LOGO



*door Bob Munniksma*

**Velen kennen reeds de programmeertaal LOGO. We hebben in ons vorige nummers al aandacht besteed aan LOGO, bv. in de uitvoering voor de Plus/4, maar er is natuurlijk ook voor de 64 een LOGO versie. We spraken toen van een interactieve taal. Gezien de oorsprong, de ontstaanswijze en de stormachtige ontwikkeling van het taaltje, mogen we als Commodore-INFO redactie die aanduiding "interactief" niet negeren.**

Immers, de ontwikkelaars van LOGO wilden een dialoog tussen computer en (jonge) gebruiker mogelijk maken. Nou, dat is volgens mij aardig gelukt.
Wat wordt er eigenlijk bedoeld met "interactief". Bij een toespraak van een spreker met luisterend publiek is er geen sprake van interactie. De communicatie is eenzijdig, van spreker tot publiek. Gaat men na een spreekbeurt met de spreker en het overige publiek in discussie, dan kan je pas spreken van interactie. Er is een wisselwerking, in welke mate dan ook, tussen de gespreksdeelnemers. Iemand zegt iets, een ander reageert en weer een ander reageert daar weer op, enzovoort. Daarbij geldt als regel, dat de ene opmerking of handeling de andere uitlokt. Bepaalde uitspraken zouden niet plaatsvinden, als ze niet waren opgewekt door andere. Dat verschijnsel is overbekend.
Nu terug naar de computer. In de meeste gevallen is er bij gebruik van een computer geen sprake van interactie. Een programmertaal zoals BASIC of PASCAL heeft wel de elementen in zich voor interactieve programmering, maar dat is misschien iets voor de gevorderde programmeurs. Ook in machine- code kan wisselwerking worden gesuggereerd. We kennen allemaal de sport-simulaties zoals Summergames, Baseball, waar sprake is van een bijna menselijk reactiepatroon tussen speler en machine.

## LISP

Een taal als LISP werd enige tijd geleden speciaal voor interactieve doeleinden ontworpen. Met LISP is het o.a. mogelijk om razendsnel ingegeven zinnen te analyseren op syntax en inhoud en adequaat te reageren. De computertaal kreeg wat bekendheid met Eliza, een psychoanalytisch programma. Eliza kon fungeren als psychiater en kon een freudiaanse analyse samenstellen naar aanleiding van verhalen van de "patient". De LISP-taal is echter nooit echt doorgebroken.
Met een LOGO-inplementatie zijn de mogelijkheden tot interactief computergebruik voorhanden. De structuur van de LOGO-taal heeft namelijk ontzettend veel weg van de LISP structuren. Een groot aantal LOGO-versies is zelfs gebaseerd op een LISP-achtige kern.
Nu beweegt interactie zich op verschillende niveaus. Op het laagste niveau mag interactie nauwelijks die naam dragen. We geven de "schildpad" de volgende opdracht:
**FD 100**
(dit is: 100 stapjes vooruit)
De opdracht wordt natuurlijk zonder problemen uitgevoerd. We zien de actie en constateren dat dit inderdaad de bedoeling was. We geven een volgende opdracht. Direct reageren is er niet bij, althans niet met enig resultaat. We moeten wachten tot de handeling is uitgevoerd. Bij een wat langere LOGO procedure, zoals het tekenen van bijv. een complexe vorm, kan het wel even duren. Met CONTROL/G kunnen we de zaak wel stopzetten, maar verder gaat de interactie niet.
Dan is er interactie mogelijk met direct merkbaar resultaat. Hier gaat het natuurlijk om. De schildpad doet iets, de gebruiker reageert en grijpt direct in het proces in. Er onstaat steeds weer een unieke situatie en elke reactie van de gebruiker is uniek. Geen vaste patronen, maar wisselwerking tussen mens en machine. Het klinkt misschien wat hoogdravend, maar dat kan je interactie noemen.

Om dit eens te demonstreren in de practijk zijn de volgende korte procedures. Om het eenvoudig te houden, blijven we uitgaan van het schildpadje. Dat is in LOGO overigens helemaal niet de enige optie voor de gebruiker! Het interactieve element is hier de LOGO-primitief READCHARACTER, afgekort met RC. Bekijk de volgende procedures maar eens:

```
TO TEKENEN

CS
LEES RC
TEKENEN
END

TO LEES :TOETS

IF :TOETS = "V FD 5
IF :TOETS = "R RT 5
IF :TOETS = "L LT 5
IF :TOETS = "S TOPLEVEL
LEES RC
END
```

De eerste procedure TEKENEN veegt het scherm schoon en start de procedure LEES. Deze procedure wacht op een toetsaanslag. Is de toets de letter V, dan gaat het schildpadje 5 stapjes vooruit. Net zo gaat het met R en L. De schildpad draait respectievelijk rechts- of linksom. Een S onderbreekt de uitvoering. TOPLEVEL stopt alle in uitvoering zijnde procedures. Alle andere toetsen worden genegeerd. Een andere toetsindruk herstart de procedure LEES. Met deze twee procedures kan de schildpad het hele scherm over wandelen, zoals de gebruiker op dat ogenblik beslist. Wederom geldt: geen vast patroon, maar de inventiviteit van het moment bepaalt het resultaat.

Nu kunnen we ook gebruik maken van de ASCII-waarde van de toetsen van het Commodore toetsenbord. Erg jammer is wel, dat de functie- en de cursortoetsen in CBM-LOGO al een bestemming hebben. Die kunnen we dus niet meer gebruiken. Maar alle andere toetsen wel. In dit voorbeeld blijven we voor het gemak uitgaan van de V, de R en de L. De S laat het programma stoppen. Aan de oorspronkelijke procedure behoeft weinig te worden veranderd. Bekijk de uitdraai maar eens. De schildpad kan nu weer over het scherm worden gestuurd. Alleen de tweede procedure wordt anders:

```
TO LEES :TOETS

IF ASCII :TOETS = 86 FD 5
IF ASCII :TOETS = 82 RT 5
IF ASCII :TOETS = 76 LT 5
IF ASCII :TOETS = 83 TOPLEVEL
LEES RC
END
```

Op deze wijze kunnen bijna alle toetsen van de C-64 een functie toegewezen krijgen.
Naast wisselwerking via het toetsenbord, kan ook de controlepoort met daarin een joystick als interactiekanaal dienen. LOGO kent voor de uitvoering een aantal primitieven. JOYSTICK leest de stand van een joystick uit en geeft een waarde afhankelijk van de stand af. JOYBUTTON kijkt of de vuurknop werd ingedrukt en geeft een 0 of een 1. Van die twee elk een klein voorbeeld. De procedures spreken verder voor zich. Voor de joystick:

```
TO STUUR :RICHTING
IF :RICHTING 0 STOP
SETH :RICHTING * 45 FD 5
END
```

```
TO JOY
STUUR JOYSTICK 1
JOY
END
```

Roep de procedure JOY aan en "loop" met de schildpad over het scherm.
Voor het gebruik van de vuurknop hoeft er weinig veranderd te worden. Procedure JOY krijgt een extra regel. Met de vuurknop ingedrukt tekent de schildpad, anders niet:

```
TO JOY
STUUR JOYSTICK 1
IF JOYBUTTON 1 PD ELSE PU
JOY
END
```

Vergeet bij het intikken van de procedures vooral de spaties niet ! Deze zijn in LOGO van essentieel belang.
Tot zover de interactie tussen mens en machine. In een komend nummer van COMMODORE-INFO zullen we eens dieper ingaan op het werken met LOGO-interactieven. •

Onze hardware-projecten krijgen veel waardering en veel reakties. Men probeert, verbetert of geeft commentaar. Deze keer weer iets voor de VIC-gebruiker.

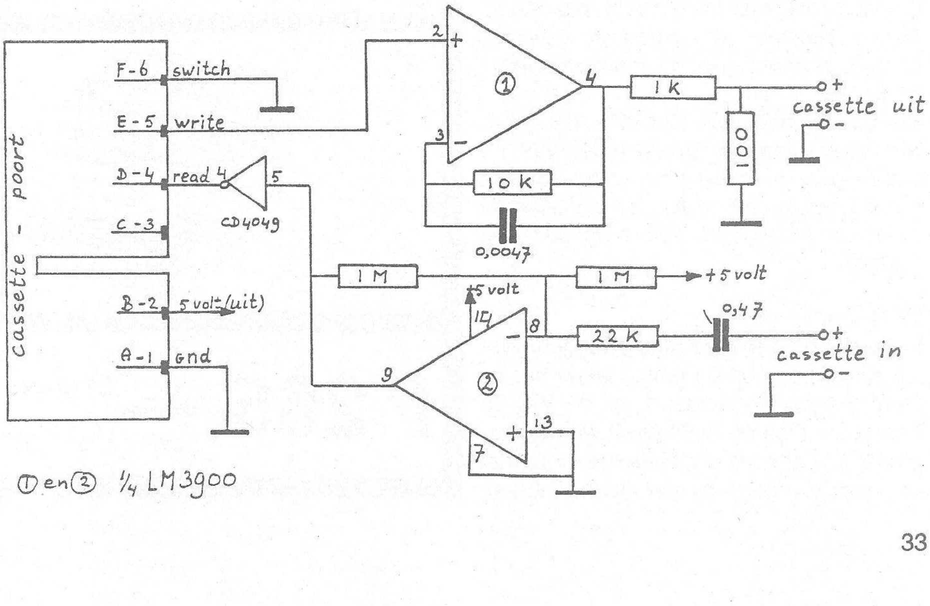
# CASSETTE-INTERFACES

**De hardware-tip van deze maand is geheel ongevaarlijk. Een cassette-interface voor de VIC-20 aangevuld met een tip gelezen in het Amerikaanse blad Commodore Computing, over het aanpassen van de SX-64 voor cassette-gebruik. Vooral dat laatste is veelgevraagd en al zijn we niet erg origineel, het belang van de SX bezitters staat hier even voorop, hun gemis aan cassettepoort is oorzaak van veel telefoontjes.**

Het is eigenlijk onzin om een cassetterecorder te kopen als je er al een hebt. Toch doen veel mensen dat. In bijna iedere huiskamer staat een stereo-set met daarin een cassette-recorder. Veel mensen hebben dan ook nog een portable recorder voor buiten, op vakantie e.d. Over het algemeen ligt die laatste de meeste tijd ongebruikt in de kast. Logisch ook: als je hifi- stereo naar muziek kunt luisteren, dan doe je dat niet mono. Of het nu gaat om klassiek of pop. Diezelfde personen kopen misschien wel een computer en ze kopen er dan meteen maar een cassette-recorder bij. Die ene recorder in de kast, wachtend op de tijd dat hij in actie mag komen, wordt geheel vergeten. Waarom een recorder gekocht als je er al een hebt ? Dat was de startvraag. Inderdaad, waarom ?
Het antwoord is duidelijk. Hoe kan je nou een "gewone" recorder aansluiten op een computer. Dat kan inderdaad niet zomaar. Toch is het goed mogelijk een gewone audiorecorder te gebruiken voor dataopslag. Het enige wat nodig is, is een verbindende schakel tussen recorder en computer. Mensen in de computerwereld noemen zo'n schakel ook wel een interface. Maar hoe kom je nou aan een cassette-interface. Er zijn twee mogelijkheden. Of je koopt er een. Elke goed uitgeruste computerwinkel kan een goedwerkende cassette-interface leveren. De prijs zal gemiddeld rond de 30 gulden liggen (zonder kabel overigens). Een andere mogelijkheid is om er zelf een te maken. Hoewel die laatste mogelijkheid natuurlijk aanzienlijk goedkoper uitvalt (hooguit zo'n 10 gulden inclusief kabel!), is er voor het maken van een interface wel wat elektronika kennis nodig. Niet zo heel erg veel, want de meeste onderdelen zitten in zogenaamde IC's. Het is een kwestie van de juiste verbindingen maken. Bij elkaar is het niet meer dan een handvol weerstanden, twee IC's, een paar condensatoren en drie diodes. Iedereen die een soldeerbout kan vasthouden, moet dit eenvoudige interface kunnen maken.

## Schema

Bekijk het schema en zie hoe simpel het is om een gewone analoge cassette-recorder digitaal te laten lezen en schrijven. En nu de praktijk. Als u een programma wil wegschrijven naar cassette, komt dit signaal op pen E-5 van de cassettepoort. Dit signaal is digitaal en wordt omgezet naar analoog met de Opamp 1. Deze zit o.a. in het IC LM3900. Andere IC's zijn ook bruikbaar (vraag de onderdelenhandelaar naar een equivalent type). Het signaal uit deze Opamp is geschikt voor de microfooningang van een recorder. Met de LED en de inverter (CD4049 of andere) kan de datastroom ook zichtbaar gemaakt worden. Als de LED brandt, gaat er een signaal naar de cassette. Het lezen van cassette gaat op exact dezelfde manier. Op de cassettepoort van de computer verwacht pen D-4 een digitaal signaal. Het analoge signaal uit de oortelefoonuitgang van de cassette wordt omgezet in een digitaal signaal in de Opamp 2 (in een LM3900 zitten er vier!). De inverter (CD4049 of andere) maakt er dan een puls van, die de computer kan gebruiken. Met De LED aan de leesleiding kan weer worden vastgesteld of er inderdaad een signaal door de



leiding gaat. De overige onderdelen zijn nodig om de signaalsterkte aan te passen en om de Opamps goed te laten werken.

## extra's

Dat is eigenlijk alles wat nodig is om een gewone recorder te gebruiken als datarecorder. Er blijven nog 4 aansluitingen over op de cassettepoort. Waar zijn die dan voor?

Als je de zaak wat luxer wil maken, dan kan dat. Op pen C-3 komt een spanning van 6,7 volt te staan, als de computer leest of schrijft. De stroom is voldoende om een relais te sturen. Met dit relais kan je dan de cassettemotor via de afstandbediening (als die er op zit) laten lopen en stoppen. Op pen F-6 zit de cassette-switch aangesloten. Deze kan met de massa worden verbonden. Als er een schakelaar in de cassette-recorder zit (dat is vaak niet het geval) kan deze pen daarop aangesloten worden. Zo kan de computer dan zien of er een toets van de cassette werd ingedrukt. Pen B-2 geeft een constante spanning van 5 volt. Met deze 5 volt voedt u de interface-schakeling op de aangegeven punten.

Pen A-1 tenslotte is de aarding van de schakeling en de signaal-leidingen. Het geheel, al of niet uitgebreid kan op een eurokaartje of iets dergelijks worden gemonteerd. Je kunt hem onderbrengen in de computer of in de recorder. In de recorder heeft m.i. zo z'n nadelen. Er is dan een speciale kabel met stekker nodig. Dit verhoogt de kosten nodeloos. Onderbrengen in de computer is veel handiger. Je kunt de bedrading aan de binnenkant op de cassettepoort solderen. De poort blijft zo ook voor andere recorders bruikbaar. Met een normale DIN-bus (bijvoorbeeld in de zij- of achterkant van de computerkast) wordt de verbinding tot stand gebracht. In de kast is genoeg ruimte voor de hele handel. Als je dan ook nog de pen-aansluitingen van de DIN-norm hanteert, kan je een gewone DIN-kabel gebruiken voor het data-verkeer. Mooier kan toch haast niet. Zorg er wel voor dat de garantie verlopen is, voor je gaat boren en schroeven in de computer.

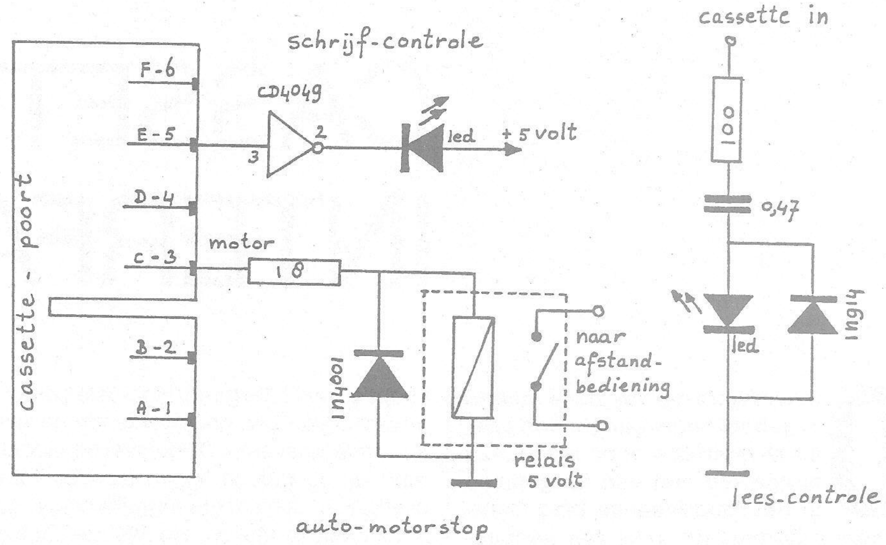
De door mij gebruikte IC's doen het goed. Andere wellicht ook. De LM3900 heeft +5 volt op pen 14, terwijl pen 7 met de massa moet worden verbonden. Bij de CD4049 moet de +5 volt op pen 1 en pen 8 is massa.

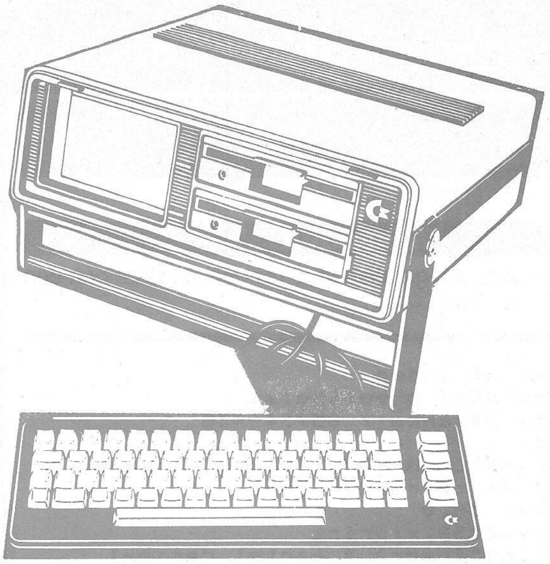
## Ook 64

Tot slot nog een paar opmerkingen. Met de genoemde componenten werkt het interface zeer bevredigend op de VIC-20 computer. Ook de C-64 geeft weinig problemen. Toch gaat er wel eens wat mis met die laatste. Vooral bij het laden. Dat kan liggen aan de andere clock-frequentie van die C-64. Ik ben zeer benieuwd naar ervaringen van andere doe-het-zelvers, die zich ook met deze materie bezighouden. Graag je reactie naar de redactie van dit blad. Misschien valt er nog iets te verbeteren.

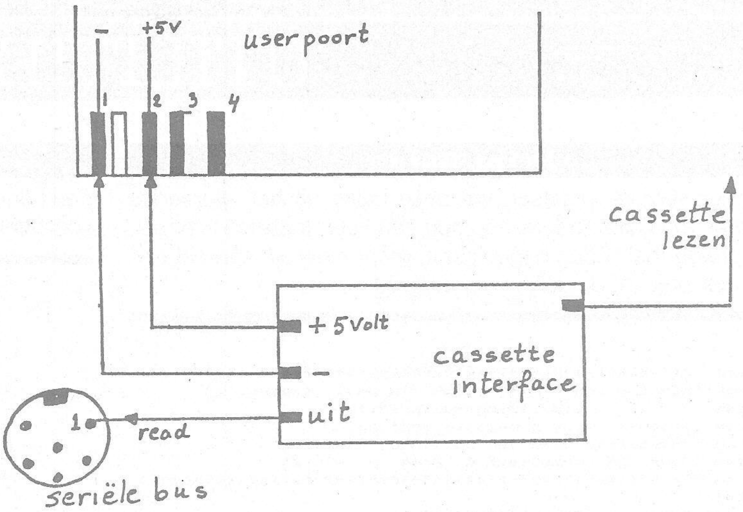
**Overzicht van de poortaansluitingen:**

A-1 Aarding
B-2 5 volt constant
C-3 6,7 volt incidenteel
D-4 Lezen van cassette
E-5 Schrijven naar cassette
F-6 Cassette-schakelaar





**Schema van aansluiting**




# Casettepoort voor de SX-64

En dan nu de SX-64. Deze compacte computer heeft schijnbaar alles. Ingebouwde disk-drive, kleurenmonitor en een bijna gewone C-64 computer. Uitbreidingspoort en gebruikerspoort zijn identiek aan de C-64. Toch is de SX-64 iets anders. Je kan niet van cassette laden, omdat er geen cassettepoort aanwezig is. Nu zal dat ook niet zo vaak nodig zijn, maar toch is het wel handig om een cassettepoort te hebben. Vooral omdat veel leuke spelletjes op cassette geleverd worden. Afgezien van het feit, dat de poort ontbreekt, is ook de KERNAL-ROM voor laadprocedures anders dan de gewone C-64. Bijvoorbeeld geeft SHIFT en RUN/STOP een laadcommando voor disk. LOAD "programmanaam",1 geeft een Illegal Device Number foutmelding.

De cassettelaad-routine zit wel in de ROM maar wordt als het ware geblokkeerd.

Een cassette READ-lijn was gemakkelijk te vinden op pen 1 van de seriële bus.

Een WRITE-lijn is niet zo van belang. Om deze "cassette-blokkade" op te heffen was het voldoende om KERNAL en BASIC RAM over te zetten naar de schaduw-RAM en daar de betreffende bits om te zetten en van bank te veranderen. Zie daarvoor de bijgevoegde listing. Eenmaal ingevoerd werkt de leesroutine voor de cassette zonder problemen.

## Recorderstroom

Het tweede probleem is de recorder zelf. Een Commodore recorder betrekt de stroom uit de computer. De SX-64 heeft onvoldoende vermogen om een cassetterecorder extra te voeden. Een zelfstandige recorder (met een interface zoals we eerder beschreven) was de enige oplossing. Het interface zelf kan wel zijn voeding uit de computer betrekken. Pen 1 en 2 van de gebruikers-poort zorgen voor respectievelijk - en +5 Volt (max. 100 mA). Direct solderen kan, maar een connector is wat netter.

In het kort:

1 LEES vanuit de seriele bus op pen 1
2 Voeding uit gebruikers-poort op pen 1 en 2
3 KERNAL en BASIC overzetten naar RAM
4 KERNAL veranderen
5 Van geheugenbank veranderen
6 Laden van cassette

De meeste programma's zullen met deze werkwijze prima laden. Bovendien hoef je niet in de computer te spitten. Alles gebeurt aan de buitenkant!

Sommige lange programma's gebruiken echter eveneens de zogeheten schaduw-RAM en vernietigen zo de KERNAL aanpassingen die we net hebben ingevoerd.

De enige oplossing hiervoor is het vervangen van de SX-64 KERNAL-ROM voor C-64 KERNAL-ROM. Een wat dure maar wel effectieve oplossing. Het uitwisselen zelf levert geen problemen op: ROM-chip eruit en ROM-chip er weer in. De SX draait prima, zij het nu met de C-64 kleuren.

Aan de slag nu en laat eens horen of het gelukt is. Wij zijn altijd bereid, anders ervaringen te horen en weer door te geven.

*B.M*

**BASIC PROGRAMMA voor KERNAL en BASIC aanpassing**

```
10 REM KERNAL EN BASIC OMZETTING VOOR SX-64
20 FOR I=57344 TO 65535:A=PEEK(I):POKE I,A:NEXT
30 FOR I=40960 TO 49151:A=PEEK(I):POKE I,A:NEXT
40 POKE 62647,123:POKE62969,95
50 POKE 64982,5 :POKE 1,5
```

# UTILITIES 1

## Print using

Op veler verzoek plaatsen we deze utility uit het allereerste nummer van Commodore-Info nog een keer. Indertijd is er een uitgebreide toelichting bijgeplaatst, maar eigenlijk spreekt een en ander ook vrij duidelijk voor zichzelf.

```
10 rem ************************************************
20 rem "Routine van 'PRINT-USING' in basic toepassing
25 rem            via string-manipulatie
30 rem  geschikt voor numerieke variabelen.
40 rem  ontwikkeld door jan bodzinga, blaricum
50 rem "voor CBM -computers &  C/64  -  VIC-20
60 rem ************************************************
70 rem
100 rem demo programma (#100-180)
105 print" Print Using (mogelijkheden)   origineel":print
110 for j=1 to 20
120 g=(tan(rnd(0))+.71)↑5*23-12  :rem randomgeneratie getal
130 i=3:d=0:gosub 1100           :rem < 1000 + 0 decimalen
140 i=4:d=2:gosub 1100           :rem <10000 + 2 decimalen
150 i=3:d=5:gosub 1100           :rem < 1000 + 5 decimalen
160 print" "g                    :rem origineel getal
170 next
180 end
1000 rem ***********************************************
1010 rem "Routine om getallen in kolommen te printen
1020 rem  op scherm of printer.
1030 :
1040 rem g = te formateren numerieke waarde
1050 rem i =  -------    aantal voor de komma
1060 rem d =  -------    aantal decimalen
1070 rem gt$= geformateerde stringvariabele
1080 rem ***********************************************
1100 x=int(g*10↑d+.5):x$=str$(abs(x))
1105 i1=i+2:i2=i+d+1
1110 gt$=right$("              "+x$,i+1+d)
1120 ifd<1then1160
1130 fory=i1 to i2:if mid$(gt$,y)<chr$(48)thennext
1140 z=y-i-1
1150 gt$=mid$(gt$,z,i+1)+left$(".0000000",z)+mid$(gt$,y)
1160 ifasc(gt$)<>32then gt$=left$("***************",i+d+1+
     (d=0))+" ":goto1180
1170 gt$=mid$(gt$,2):x$="-":ifnotsgn(g)thenx$=chr$(160)
1175 gt$=gt$+x$
1180 printgt$;"  ";
1190 return
```

## Wartaal 64

Dit lijkt enigszins op het vorige programmaatje, alleen maakt dit van alle tekst op het scherm complete wartaal. Deze routine zou aan het einde van een spel of programma kunnen dienen om het af te sluiten.

```
 1 rem wartaal  /  cbm-64
 2 rem door edwin vd elst
 3 rem r'dam /010-205844
 4 rem
10 for t=1024 to 2023
20 a=peek(t):ifa>27then50
30 k=a+1:ifk=27thenk=1
40 poke t,k
50 next

*** EINDE LISTING ***

wartaal
regel 1    51       regel 20   134
regel 2    12       regel 30   25
regel 3    254      regel 40   98
regel 4    143      regel 50   130
regel 10    185     ready.
```

## Afronden

Een handige routine om getallen af te ronden, er staat ook een voorbeeld bij.

```
 1 rem afronden
 2 rem door p smithuis
 3 rem barchem / 05734-438
 4 rem
10 print"[CLR-HOME][HOME][2xneer]
   * af te ronden getal : ";
20 open 50,0
30 input#50,x:print
40 print"[neer] * aantal decimal
   en[3xspatie]: ";
50 input#50,y:y=int(y+0.5)
60 close 50:print
70 print"[2xneer] * afgeronde ge
   tal[4xspatie]:";(int(x*10↑y+0
   .5))/10↑y
80 wait198,1,0:goto 10

*** EINDE LISTING ***

afronden                  regel 30   64
regel 1    116            regel 40   192
regel 2    137            regel 50   79
regel 3    127            regel 60   216
regel 4    143            regel 70   171
regel 10    151           regel 80   17
regel 20    96            ready.
```

## Foute Boel

Wat hier gebeurt blijft een verrassing tot u het programma runt.

```
 1 rem foute boel / cbm-64
 2 rem door p bos
 3 rem meedhuizen / 05963-565
 4 rem
40 code=int(rnd(1)*32)
50 poke781,code  :sys42039

*** EINDE LISTING ***
```

## Binnenstebuiten 64

Hiermee worden alle tekens op het scherm in een keer in REVERSE gezet.

```
 1 rem binnenstebuiten / cbm-64
 2 rem door edwin vd elst
 3 rem r'dam / 010-205844
 4 rem
10 for t=1024 to 2023
20 a=peek(t):ifa=32then50
30 k=a+128:if k>255thenk=a-128
40 poket,k
50 next

*** EINDE LISTING ***

binnenstebuiten
regel 1    148       regel 20   131
regel 2    12        regel 30   11
regel 3    254       regel 40   98
regel 4    143       regel 50   130
regel 10    185      ready.
```

# UTILITIES 2

## Basic Keywords

Dit hulpprogramma van Markjan vd Torre geeft aan een groot aantal toetsen een extra betekenis en wel die van Basic Keywords, Door het indrukken van de combinatie (SHIFT) en een letter komt er in één keer een compleet Basic keyword op het scherm. Zo zet bijvoorbeeld (SHIFT + A) het woord ASC op het scherm en (SHIFT + C) levert CHR$ op. Maar er zijn meerdere toetsen met verkorte commando's, probeer maar eens wat letters. Er zit een eigen controlegetal in dit programma ingebouwd en u moet er aan denken, dat dit wordt weggezet vanaf adres 49152 en daar kunnen ook andere programma's wel komen. Maar verder is dit voor de echte programmeur een soort uitbreiding van de funktietoetsen en gaat het lijken op de manier, waarop bv. bij de Sinclair de Basic commando's worden aangeroepen.

```
   1 rem basic keywords / cbm-64
   2 rem door markjan van der torre
   3 rem zevenhuizen
   4 rem
  20 t=0:forx=49152to49289:reada:t=t+a:
     pokex,a:next
  30 ift<>20485thenprint"fout in data !
     !!":end
  40 sys49152
 100 data 120 , 169 , 13 , 141 , 20
 101 data 3 , 169 , 192 , 141 , 21
 102 data 3 , 88 , 96 , 165 , 212
 103 data 240 , 3 , 76 , 49 , 234
 104 data 165 , 215 , 201 , 219 , 176
 105 data 247 , 201 , 193 , 144 , 243
 106 data 56 , 233 , 193 , 170 , 169
 107 data 0 , 133 , 198 , 169 , 158
 108 data 133 , 251 , 169 , 160 , 133
 109 data 252 , 189 , 111 , 192 , 170
 110 data 160 , 0 , 10 , 240 , 16
 111 data 202 , 16 , 12 , 230 , 251
 112 data 208 , 2 , 230 , 252 , 177
 113 data 251 , 16 , 246 , 48 , 241
 114 data 200 , 177 , 251 , 48 , 17
 115 data 8 , 142 , 255 , 192 , 230
 116 data 198 , 166 , 198 , 157 , 119
 117 data 2 , 174 , 255 , 192 , 40
 118 data 208 , 234 , 230 , 198 , 166
 119 data 198 , 41 , 127 , 157 , 119
 120 data 2 , 169 , 20 , 141 , 119
 121 data 2 , 230 , 198 , 76 , 49
 122 data 234 , 198 , 169 , 199 , 134
 123 data 128 , 129 , 161 , 144 , 133
 124 data 137 , 141 , 200 , 202 , 130
 125 data 159 , 151 , 194 , 201 , 196
 126 data 163 , 183 , 197 , 131 , 135
 127 data 140 , 158 , 127

*** EINDE LISTING ***
```

```
basic keywords
regel 1     1          regel 30    214
regel 2     19         regel 40    163
regel 3     234        regel 100   194
regel 4     143        regel 101   155
regel 20    135        regel 102   118

regel 103   111        regel 116   76
regel 104   52         regel 117   157
regel 105   50         regel 118   58
regel 106   11         regel 119   12
regel 107   218        regel 120   152
regel 108   48         regel 121   118
regel 109   53         regel 122   73
regel 110   88         regel 123   50
regel 111   190        regel 124   30
regel 112   204        regel 125   58
regel 113   209        regel 126   56
regel 114   208        regel 127   168
regel 115   207        ready.
```

## Disk Name Change 1541

Hiermee kan de naam van een disk zelf worden veranderd, zonder dat er problemen komen met sector ID etc. Dus na kopiëren kan nu de nieuwe disk een andere naam krijgen, hetgeen de overzichtelijkheid van uw diskotheek ten goede kan komen. Voor de C-16 en VIC regel 10 weglaten.

```
   1 rem disk name change
   2 rem door wido van krieken
   3 rem naarden / 02159-43202
   4 rem
   5 clr
  10 poke53280,0:poke53281,0
  20 print"[CLR-HOME][2xneer][2xrechts
     ][GROEN] disk name change"
  30 open15,8,15,"i":open1,8,2,"#"
  40 print#15,"u1:";2;0;18;0:rem t
     rack 18 sector 0
  50 print#15,"b-p:";2;144:rem 'po
     sition'
  60 a$="[16xspatie]":ifa=1then80
  70 input"[neer]wat is de nieuwe
     disk naam ";b$
  80 b$=left$(b$+a$,16):fori=1to16
     :print#1,mid$(b$,i,1);:nexti
  85 print#15,"u2:";2;0;18;0
  90 print:close15:close1:ifa=0then
     100
  95 return
 100 a=1:gosub20:input"[2xneer] no
     g een schijf ";i$:ifi$<>"j"then
     sys64738
 110 clr: goto20
```

```
disk name change
regel 1     129        regel 50    52
regel 2     228        regel 60    83
regel 3     224        regel 70    161
regel 4     143        regel 80    237
regel 5     156        regel 85    22
regel 10    37         regel 90    4
regel 20    220        regel 95    142
regel 30    19         regel 100   2
regel 40    188        regel 110   193
```

# Print-out

*Met o.a. VIC 20 Lotus - Woestijnrace - Lotto - Crazy Sprite 64 - C-16 Data base*

**Er is weer een bonte variatie aan listings in dit nummer, van extreem kort tot zeer lang.**
**Ook bezitters van de VIC 20 en C-16 komen uitgebreid aan hun trekken. Wij zijn vooral blij een goed database programma voor de C-16 te kunnen plaatsen. Het spreekt vanzelf dat wij ons voor nieuwe listings voor de C-16 aanbevolen houden.**
**Ook C-64 programma's blijven natuurlijk welkom, maar we zijn nog steeds druk bezig om de enorme berg van de prijsvraaginzendingen te verwerken. Alles bij elkaar zijn er bijna 500 programma's ingestuurd, zodat de uitslag van de wedstrijd nog even op zich zal laten wachten.**

## INHOUD VAN DIT LISTINGDEEL



## Kleurshow

Een fraaie one-liner met kleurige bolletjes, die door het beeld suizen. Nuttig als "illustratie" in een ander programma.

```
 1 rem kleurshow / cbm-64
 2 rem door edwin vd elst
 3 rem r'dam / 010-205844
 4 rem
10 fort=1to20:print"Q";spc(r/2);
   :next:r=r+1:poke646,r:ifr<255
   then10


   regel 1    235
   regel 2    12
   regel 3    254
   regel 4    143
   regel 10    160

   ready.
```

## VIC 20 Hoger-lager

Een vrij bekend spel, waarbij de speler moet raden of de volgende kaart hoger of lager zal zijn.

```
0 rem hoger lager / vic-20
1 rem door p vd kroon uit amste
  rdam 020-329895
2 rem kies door toetsen -h- -l-
   of -p- voor hoger, lager of
  passen; indien men
3 rem past krijgt men evenveel
  punten als kaarten; indien er
   9 kaarten goed
4 rem voorspelt zijn krijgt men
   20 punten. veel succes
5 dima$(52),b$(36),c$(4),d$(13)
  :forx=1to4:readc$(x):next:for
  x=1to13:readd$(x)
6 next:forx=1to4:fory=1to13:a$(
  (x-1)*13+y)=c$(x)+d$(y):nexty
  ,x:readb$(1)
 7 readb$(2):forx=3to5:b$(x)=b$(
   x-1):next:readb$(6):readb$(7)
   :b$(12)=b$(7)
 8 forx=8to11:b$(x)=" ":next:for
   x=13to30:b$(x)=b$(x-6):next:read
   b$(31):readb$(36)
 9 forx=32to35:b$(x)=b$(x-30):next
   :poke53281,1:poke53280,1
10 print"[CLR-HOME][ZWART]speler
   1:"a1tab(20)"speler2:"a2:print
   "[RVS-aan]h[RVS-uit]oger[10xspati
   e][RVS-aan]l[RVS-uit]ager";
11 print"[9xspatie][RVS-aan]p[RVS-ui
   t]as":z=52:v=2:h=0:b$(5)="-":
   b$(19)="|":gosub50:gosub55
12 forq=1to8:gosub61:ifs=1thenv=
   v+2:gosub50:e$=c$:gosub55:gosub
   65:ifr=1thennext
13 ifa$="p"thena1=a1+q
14 ifq=9thena1=a1+20
15 a$="":v2=0:h=0:gosub60:print"
   [ZWART]speler1:"a1
18 v=2:h=20:b$(5)="-":b$(19)="|"
   :gosub50:gosub55
19 forq=1to8:gosub61:ifs=1thenv=
   v+2:gosub50:e$=c$:gosub55:gosub
   65:ifr=1thennext
20 ifa$="p"thena2=a2+q
21 ifq=9thena2=a2+20
22 a$="":v2=0:h=20:gosub60:print
   "[ZWART]speler2:"a2
25 goto10
28 ifa$="p"thena2=a2+q
49 end
50 t=int(rnd(0)*z+1):h$=a$(t):for
   x=ttoz-1:a$(x)=a$(x+1):next:a
   $(z)=h$:z=z-1
51 return
55 v2=v:forx=1to6:v2=v+x:gosub60
   :fory=1to6:print"[ZWART]"b$((
   x-1)*6+y);:nexty,x:v2=v+2
56 h=h+1:gosub60:printh$:b$(5)="
   ┴":b$(19)="┤":c$=right$(h$,1)
   :return
60 poke211,h:poke214,v2:sys58640
   :return
61 s=0:geta$:ifa$=""then61
62 ifa$="h"ora$="l"thens=1:return
63 ifa$="p"thens=0:return
64 goto61
65 r=1:d$=right$(h$,1):gosub70:d
   =x:d$=e$:gosub70:c=x
66 ifa$="h"andd<cthenr=0
67 ifa$="l"andc<dthenr=0
```

```
68 return
70 ifd$="0"thend$="10":forx=1to1
   3:ifd$=d$(x)thenreturn
71 forx=1to13:ifd$=d$(x)thenreturn
72 next:return
99 data "[ROOD]S","[ROOD]Z","[ZWART]
   A","[ZWART]X",2,3,4,5,6,7,8,9
   ,10,b,q,h,a,"U","-","I","|","
   J","K"

*** EINDE LISTING ***
```

## Raden maar VIC 20

Een woordenraadseltje, waarbij de computer een ingegeven woord helemaal door elkaar haalt. Iemand, die het niet gezien heeft, moet het dan weer raden en daarmee is dit een soort gezelschapsspelletje.

```
  1 rem raden maar / vic 20
  2 rem door christophe breysens
  3 rem uit groot-bijgaarden
  4 rem
 40 lt$="000100"
 50 poke36879,110
 60 printchr$(14):printchr$(8)
 70 print"[CLR-HOME][WIT]Diegene
    die het woord moet raden,verl
    aat nu de kamer."
 80 print"[neer]Het woord mag max
    imum uit [RVS-aan]20[RVS-uit]
     letters[8xrechts]bestaan."
 90 print"[neer]U moet het woord
    zo[3xrechts]snel mogelijk vin
    den."
100 print"[neer]De laagste tijd b
    lijft bewaard..."
110 print"[2xneer]Geef nu het woo
    rd:[neer]"
120 inputw1$
130 iflen(w1$)>20thenprint"max. 2
    0 letters":goto110
140 print"[CLR-HOME][HOME]┌-------
    -----------------┐";
150 print"|[20xrechts]|";
160 print"└--------------------------┘"
170 restore
180 print"[HOME][neer][rechts]";
190 fora=1to20
200 readb
210 printmid$(w1$,b,1);
220 next
230 ti$="000000":c=0:w2$=""
240 print"[HOME][5xneer]Tijd:";
250 printmid$(ti$,3,2);":";right$
    (ti$,2)
260 geta$:ifa$=""then240
270 c=c+1
280 ifa$=mid$(w1$,c,1)then350
290 poke36878,15
300 poke36876,200
310 fort=1to200:next
320 poke36876,0
330 c=c-1
340 goto260
350 poke36878,15
360 fore=200to220
370 poke36876,e
380 fort=1to20:next
390 next
400 poke36876,0
410 w2$=w2$+a$
420 print"[HOME][10xneer]";w2$
430 ifw2$=w1$thent$=ti$:goto450
440 goto260
450 poke36878,15
460 forf=1to16
470 reads
480 poke36876,s
490 poke36879,s
500 fort=1to100:next
510 next
520 poke36876,0
530 poke36879,110
540 ift$<lt$thenlt$=t$
550 print"[CLR-HOME]Proficiat!!!"
560 print"[2xneer]Jouw tijd was:"
570 printmid$(lt$,3,2);":";right$
    (t$,2)
580 print"De laagste tijd is:"
590 printmid$(lt$,3,2);":";right$
    (lt$,2)
600 print"[3xneer]Wil je nog eens
    ?"
610 geta$:ifa$=""then610
620 ifa$="j"then50
630 poke36879,27:print"[CLR-HOME]
    [BLAUW]":end
640 data13,4,10,6,19,14,3,15,5,1,
    7,20
650 data16,8,2,18,11,9,12,17
660 data223,227,230,231,234,236,2
    38,239
670 data239,238,236,234,231,230,2
    27,223

*** EINDE LISTING ***
```

## VIC 20 Lotus

Wel veel tikwerk, maar dit is een mooi plaatje van zo'n dure sportauto. Hetgeen wel bewijst, dat ook de VIC op grafisch gebied nog wel wat kan.

```
 0 rem lotus esprit / vic 20
 1 rem door remco zeeuw uit ijmu
   iden
 2 poke52,28:poke56,28:clr
 3 fori=7168to7679:pokei,peek(i+
   25600):next
 4 fori=7168to7615:reada:pokei,a
   :next
 5 poke36869,255
 6 print"[BLAUW][CLR-HOME][9xneer]"
   ;:poke36879,25
 7 printtab(8)"@abcdefgh"
 8 printtab(5)"ijklmnopqrstu"
 9 printtab(4)"vwxyz[£]↑← !#$"
10 printtab(4)"%&'()*+,-./01"
11 printtab(6)"234567"
12 goto12
13 data000,000,000,000,000,003,0
   12,048
14 data000,001,006,056,192,000,0
   00,000
15 data007,248,000,000,000,000,0
   00,000
16 data255,001,000,000,000,001,0
   02,004
```

Het woord Robot heeft nog altijd een mystieke klank. B
verhalen zijn de onvermoeibare werkers of agressieve vech
echt aan zo'n "Werker" hebt blijft voor velen onduidelijk. In
gaat Ulco Schuurmans in op het nut van robots en hun gebr
Huis- en Personal Computers. Helaas blijft de toepassing va
res nog beperkt tot wat A/D (analoog/digitaal) interfaces, to
en tot het bouwen van bewegende constructies met Fish
terwijl er misschien ook al lezers zijn, die robotarmpjes aan h
dit gebied toch heel interessant is, een overzicht van Rob

# ROBOTS: MEER ALLEEN MAAR SPEELGOED!

**door Ulco Schuurmans**

**Robots in de vorm van bewegende constructies zijn er al bijna net zo lang als de werktuig- en speelgoedbouw. Een uurwerk met bewegende figuurtjes, een orgel, er zijn erg veel voorbeelden te geven. Maar gaan we verder dan het leuke, het recreatieve en praten we over robots die iets nuttigs voor ons kunnen doen, dan praten we over enige tientallen jaren.**

De eerste Robots waren volledig mechanisch van aard. Handbesturing was onmisbaar om de Robot zijn werkzaamheden te laten verrichten. Ook nu bestaat een groot deel van de werkende robotgarde uit door mensen bestuurde machines, maar langzamerhand worden steeds meer zelfstandige taken ingebouwd of aan een externe computer supervisor overgedragen. Bekend is bijvoorbeeld de toepassing van las- en lakrobots in de autoindustrie. Voorgeprogrammeerde lopende-band robots construeren vrijwel zonder menselijk ingrijpen in korte tijd grote aantallen auto's. Ondanks het feit dat dergelijke Robots dikwijls doof, blind en op hun kraan- en montage-armen na onbeweeglijk zijn, gaat het toch om zeer gecompliceerde machinerie, maar zonder veel gelijkenis met de mensen.

## Andere robotvormen

De Cyborgtechniek is gebaseerd op geheel andere uitgangspunten.

**Cyborg** is een samentrekking van Cybernetica (= stuurmanskunst) en organisme, een wat misleidend begrip. In de praktijk bedoelt men de koppeling van mechanische (robot) onderdelen aan het menselijk lichaam. Kunstarmen en benen van zeer goede kwaliteit zijn al op beperkte schaa leverbaar. Ook kunstharten en zintuigen zullen binnenkort niet vreemd meer zijn. Experimenten om biologische "intelligentie" in mechanische systemen in te bouwen hebben tot nu toe alleen maar bedroevende resultaten opgeleverd.

Als laatste de **Androiden** die ook wel tot de Robots gerekend wordt. Het gaat hier om een "organische kunstmens", geschikt voor robotwerk. Praktische voorbeelden hiervan kennen we niet, het is een concept uit de Science-Fiction wereld.

## Werking van een Robot

Een mechanische Robot bestaat uit de volgende componenten:

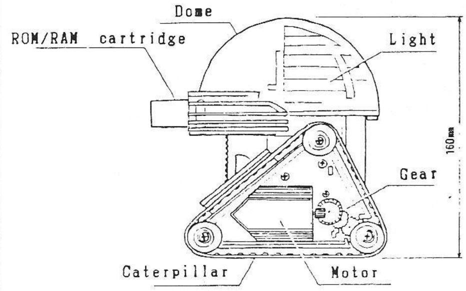
❶ Een besturingssysteem, dat zeer gecompliceerd of bijzonder eenvoudig kan zijn. In het eerste geval wordt het besturingssysteem voortdurend geïnformeerd over de gang van zaken bij de verrichte

## Movit-serie van Owi

Het Amerikaanse bedrijf OWI brengt een uitgebreide serie Movit Robotjes uit waarvan enkele met de C64, IBM PC en Apple II's te besturen zijn. De Movit Educational Electronic Robot Kits beschikken over 4KB RAM-geheugen en zijn bedoeld voor technisch geinteresseerde hobbyisten of het verrichten van kleine onderwijs experimenten.

Door middel van 7 besturingsfunkties kan de gebruiker inzicht verwerven over hetgeen er allemaal komt kijken bij het besturen van een robot.

Een kleine greep uit de Movit-serie : **MEMOCON-CRAWLER** gebruikt zijn BAB CMOS static RAM 256 x 4 bit geheugen voor de funkties : vooruit, links/rechts, pauze, zoemen en knipperen.

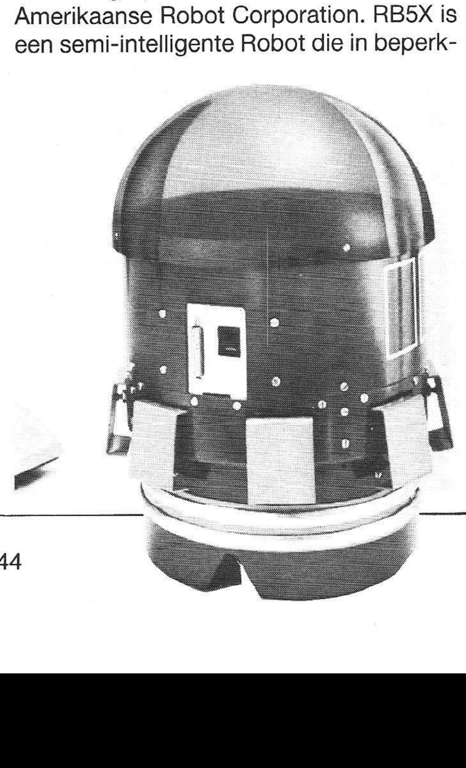



De OWI prijst Memocon Crawler aan voor Basic experimenten. De eenvoudige **PIPER-MOUSE** reageert op supersone fluitsignalen door naar links/rechts te draaien en te stoppen of te gaan rijden.

**TURN BACKER** reageert op de kreet "Look out" (=kijk uit) door naar links uit te wijken. Alle MOVIT robotjes komen als gemakkelijk te monteren bouwpakket en lopen op standaard AA 15V batterijen. Alleen de genoemde MEMOCON CRAWLER MV-918 valt zinvol op uw hobbycomputer te programmeren via een optionele interface kit.

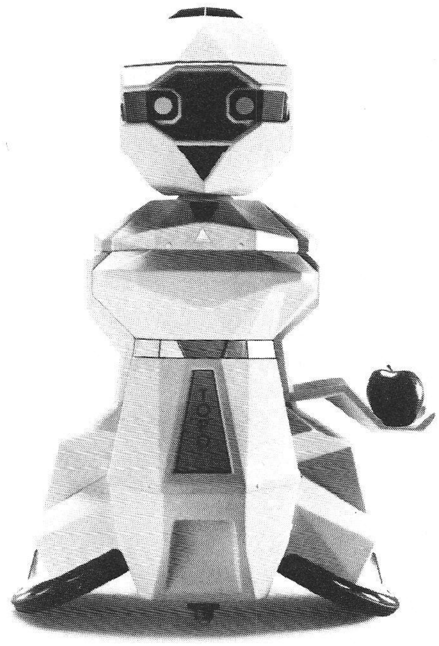
## De Robot Corporation

Meer mogelijkheden biedt de **RB5X** van de Amerikaanse Robot Corporation. RB5X is een semi-intelligente Robot die in beperk-



ɜkend uit science fiction
:machines, maar wat je nu
dit en volgende artikelen
ɹiksmogelijkheden op de
ı robots bij de Commodo-
: het aansturen van treintjes
ɜr-Technik® bouwdozen,
ɹn 64 koppelden. Omdat
ɔtica-nieuwtjes.

**DAN**



werkzaamheden en kunnen via terugkoppeling noodzakelijke korrekties uitgevoerd worden. In het tweede geval is het slechts mogelijk een gegeven opdracht in zijn geheel uit te voeren zonder aanpassing aan optredende veranderingen.

❷ Diverse uitvoerende systemen; deze bestaan gewoon uit door electromotoren aangedreven mechanische onderdelen.

❸ Signaleringssysteem; deze informeren het besturingssysteem over de geboekte resultaten en eventueel optredende veranderingen in de werksituatie. Het signaleringssysteem ontbreekt dikwijls op de eenvoudiger robottypen.

❹ Krachtbron, in verreweg de meeste gevallen accu's. Sommige robots worden extern gevoed.

❺ Een eigen- of extern programmeringssysteem. Eenvoudige robots zijn vaak al (ROM-) voorgeprogrammeerd.

❻ Een funktionele behuizing, afhangend van het doel en de omstandigheden waaronder een robot moet werken.

❼ Een communicatie-systeem, bijvoorbeeld optische signalen of echte spraak.

## De plaats van de robot

In onze steeds verder automatiserende maatschappij neemt het belang van de Robot snel toe. Ze rukken op omdat ze goedkoper of beter kunnen werken of in omstandigheden, waar dat voor mensen bezwaarlijk is. Momenteel treffen wij Robots o.a. op de volgende terreinen aan:

○ in voor de mens gevaarlijke of onmogelijke werksituaties;
○ in de produktiesectoren waar continu een grote hoeveelheid standaardwerk verricht moet worden. De Robot is immers onvermoeibaar en verveelt zich nooit;
○ als huishoudhulpje, eigenlijk nog niet echt een serieuze toepassing, maar wie weet moeten wij er binnenkort er ook aan geloven;
○ als educatief speelgoed, bijvoorbeeld de LOGO-schilpad;
○ als militair wapen, daar doet men uiteraard heel geheimzinnig over, maar de plichtsvervulling is boven elke twijfel verheven;
○ als reclame-object;
○ als receptionist, bestellingopnemer of voorlichter;
○ in bewakings- en signaleringsfunkties, o.a. inbraak- en branddetectie;
○ als thuis-, film- of theaterproduktie-entertainer.

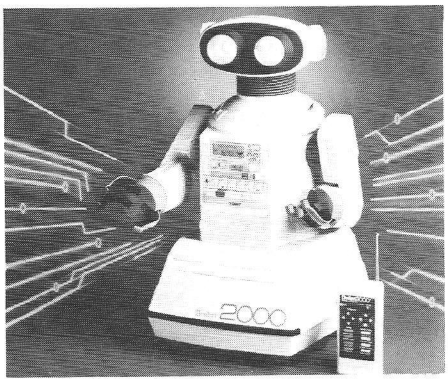


te mate van de opgedane ervaringen kan leren. Aangesloten op een standaard RS-232 interface kan RB5X door zijn eigenaar getraind, aangepast en bestudeerd worden. Standaard beschikt het ronde Robotje daartoe over 8K uit te breiden met nog eens extra 16K RAM.

Behalve de ingebouwde tactiele (naderings) sensors kan de RB5X ook met sonar- en fotocellen uitgerust worden om de omgeving te verkennen. Verder zijn verkrijgbaar: een robotarm, data telemetrie (waarmee ook een andere RB5X geinstrueerd kan worden!), software testprogramma's en signaallichtjes.

## Tomy

De Tomy Corporation brengt nu voor hun al wat langer bestaande Omnibot 2000, Omnibot en Verbot huis-amusementsrobots het software pakket ROBOTLINK (TM) uit. Programmeren van de genoemde robots is nu mogelijk met Atari, Commodore, IBM en Tandycomputers. Het Robotlink-programma stelt de gebruiker via een speciale kabelverbinding in staat programma's te saven, laden en te mixen. Hierdoor krijgen de robotjes meer bewegings- en daarbij zelfs (charade- en adventure) spelmogelijkheden. Ingebouwd is al **Spin the Robot** en aangekondigd zijn **Land of Imagination** (een klank-, licht- en bewegingsspel voor kinderen vanaf 3 jaar) en **Robot Basketball**.



De Tomy range bestaat nu uit:

- De Omnibot, een voorprogrammeerbaar robotje.
- De Omnibot 2000, het topmodel met de volgende mogelijkheden: serveren, voorwerpen terugbrengen, het begroeten van bezoekers, een volledig programmeerbare rechter robotarm met gewrichten en drieledige hand, een persoonlijke werkfunktie en volledige PC/HC-programmeerbaarheid. Bewakingsfunkties behoren eveneens tot de funktiemogelijkheden.
- De Verbot, een op afstand bediend robotje dat op gesproken commando's, waaronder glimlachen , reageert.
- Chatbot, kan een 30 seconden durende gesproken boodschap overbrengen.

En de echte speelgoedrobotjes Hootbot, Sportbot, Dinbot en Flipbot.

## Androbot

De **TOPO**-robot van Androbot (opgericht door Nolan Bushnell, de stichter van Atari) is in feite een stuk mobiele en bestuurbare randapparatuur van de computer. De huiscomputer/PC funktioneert als brein en geheugen. De TOPO voert de gegeven opdrachten uit. De besturing verloopt via een geavanceerd infrarood interface-systeem dat TOPO's microprocessor instrueert. **TopForth** is de gebruikte programmeertaal, waarmee veel mogelijk is: educatieve spellen, spreken, zelf programmeren en eenvoudige bewegingsinstrukties via het toetsenbord of joystick. TOPO is voorzien van een interne uitbreidingsbus waarop nieuwe electronica en sensoren aansluitbaar zijn.
**B(rain) o(n) B(oard) BOB** is een volledige Personal Robot van Androbot Inc. BOB's intelligentie zit hem in de Intel 8086 processors gekombineerd met drie megabytes-geheugencapaciteit. De robot beweegt botsingloos door het huis, kent meer dan 100 woorden, spreekt met een menselijke stem en is voorzien van ultrasone (voor het ontwijken van dode voorwerpen)- en infrarode (voor het opmerken en volgen van mensen en dieren)-sensoren. De uitbreidingsmogelijkheden zijn legio met commercieel verkrijgbare of zelfgemaakte hard- en software. Zo zijn er bijvoorbeeld CPU en spraakherkenning of een speciale Androwagen voor het transport van boeken, drankjes e.d. De programmeertaal Androbot Control Language (ACL) maakt gebruik van zeer gebruikersvriendelijke commando's zoals **Turn head to right** (=draai hoofd naar rechts). Via een seriële interface koppelt het pakket vrijwel elke hobbycomputer aan BOB.

Een ander interessant Androbot product is de**AndroMan VIDEOROBOT**. Hiermee worden videospelletjes driedimensionale werkelijkheid. De 12 inch geminiaturiseerde Androbot bestuurt via een infrarood joystick waarbij de actie beurtelings van het 3D- robotmilieu naar het videoscherm en omgekeerd schiet. Spraaksynthese geeft in de vorm van waarschuwingen en pluimpjes een extra charme aan het videospel.

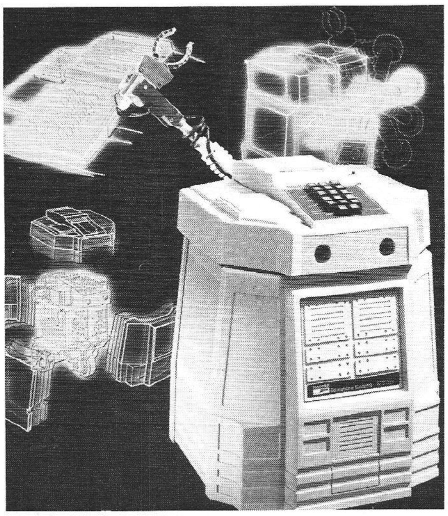
## De Personal Robot.

Een personal robot is een robot die thuis, of op school een educatieve, huishoudelijke, administratieve of vermakende funktie vervult. Kortom de "ideale huis- of schoolvriend".
In de VS heeft zich al een komplete huisrobot markt ontwikkeld. Bewakings-,huishoud- en educatieve robot/computers zijn er te kust en te keur voor een redelijke prijs. Om enig onderscheid in dit overvloedige aanbod te kunnen maken kunnen we de Robots in twee sorrten indelen: de Robots bestuurd met een externe huis- of personal computer en de zgn. "Brein Aan Boord" (BAB) Robots met ingebouwde computer.

## Robots in het onderwijs

Vele robots worden als educatief aangeprezen. Bij de goedkopere modellen vallen de onderwijstalenten helaas nogal tegen. De duurdere Robots bieden mogelijkheden om (BASIC-) programmering, artificiele intelligentie, cybernetische mechanica en taalstructuren d.m.v. experimenten te bestuderen. Robots hebben een soort natuurlijke charme die vooral kinderen aanspreekt. Spelend leren is daarom zeker een robotverdienste.
Het Engelse bedrijf Valiant maakt voor LOGO-besturing de **Valiant** Schildpad. De bedoeling van de ontwerper Seymour Papert is dat het kind de computer via de bewegingen van de robotschildpad leert programmeren. Interfacing is mogelijk met Apple, BBC, Commodore, IBM en Spectrum-computers. De schildpad zelf beweegt als een echte plotter over een vel tekenpapier en maakt zo figuren met Berol fibre tip-pennen. Een aanzienlijk duurdere-,maar ook veelzijdiger uitvoering van de "Schildpadtechniek" is de **Penman Plotter** voor BBC, Apple II of IBM PC. Ook hier wordt een LOGO-achtige stuurtaal gebruikt met bv. Penup en Pendown voor plottergebruik.





## ZELFSTANDIGE ROBOTS

Deze groep heeft de intelligentie ingebouwd en kan dus als "Stand-alone" of "move-alone" aan de gang.
De ook in ons land bekende **HERO I** van Zenith is een voorbeeld van een robot die over ingebouwde programmeerbare computerbesturing beschikt. Koppeling aan een PC is niet nodig om de HERO I tot actie aan te zetten. Deze Personal Robot, hoewel de toepassingen minder in het persoonlijke als wel in het onderwijsvlak liggen, beschikt over indrukwekkende specificaties:

**Sensoren:**
1 Geluiddetectie van 200-5000 Hz omnidirectioneel.(256 nivo's).
2 Lichtdetectie over het gehele zichtbare spectrum. Sensorhoek 30 graden.
3 Ultrasone 35 KHz detectie.

**Spraaksynthese:**
Genereert 64 basisgeluiden waarmee de menselijk stem en diverse geluidseffecten gesimuleerd kunnen worden.

**Tijd:**
Voor vier jaar vooruit instelbare kalenderklok.

**Overige funkties:**
- 350 graden draaibaar hoofd met LED-display, hexadecimaal toetsenbord en de bovengenoemde sensoren.
- multifunktionele robotarm.
- 6808 Microprocessor met 7 operating modes:
  1) executive, 2) utility, 3) program, 4) repeat, 5) manual, 6) learn, 7) sleep.

De **HERO I** van Zenith is in de eerste plaats bedoeld als educatieve Robot, maar is ook voor lichte industriële en huiselijke werkzaamheden geschikt. ●

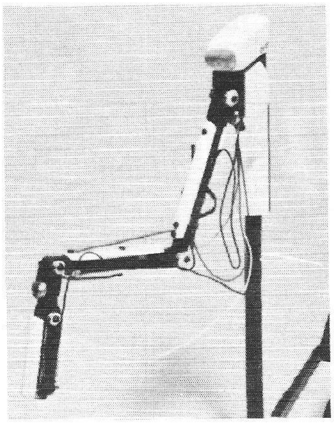
## Hubo

De butler-achtige **HUBOT** van Hubotics is volgens de fabrikant de optimale huisrobot. Het opsporen van inbrekers en het begin van brand wordt automatisch gevolgd door het bellen van politie of brandweer. Verder kan de Hubot bedienen, gasten begroeten, 1200 woorden verstaan en overhoren, voorwerpen verplaatsen, stofzuigen, wekken, de financiele administratie beheren, stereo- cassettes afdraaien en Atari 2600-systeem video-spelletjes spelen.
Deze "ideale" computerhuisvriend beschikt over een 64K PC en een CP/M besturingssysteem. Aangekondigd is als accessoire een robotarm.



## Biowire

In Japan maakt Toki Corporation **Biometal en Biowire** Robots. Het Biowire is een zelf herinnerend veermetaal. Komt het draadje onder stroom te staan dan zet het uit. Het uitschakelen van de stroom doet Biowire tot vorige veerlengte teruggaan. Een ideaal materiaal om Robots voor eenvoudige opdrachten mee te bouwen.



# MISSERS

**Ook in de vorige nummers van Commodore-Info is helaas het één en ander misgegaan. Hoewel de meeste dingen niet zo ernstig waren, zetten we hier de gemaakte fouten recht. Gelukkig komen er steeds minder foutmeldingen bij ons binnen.**

Dat kan niet anders betekenen, dan dat er steeds nauwkeuriger wordt gewerkt. En met een oplage die tegen de 50 000 loopt, krijg je zelfs bij de kleinste misser direkt tientallen telefoontjes. En we zijn natuurlijk geen adviesbureau voor computerproblemen, maar dat heeft u ondertussen wel begrepen. Heeft u echt een misser ontdekt, probeer dan ook de oplossing erbij te zoeken, daar hebben we iets aan. Maar nog altijd is het zo, dat zo'n 90 tot 95 procent van alle 'syntax- errors' die we binnenkrijgen uiteindelijk te wijten is aan fouten die tijdens het overtypen door de lezers gemaakt. Ook blijkt, dat even wachten wel helpt bij moeilijker grafische programma's en dat goed lezen niet misplaatst is. In het programma Infoteek bijvoorbeeld stond duidelijk dat **S** bedoeld was voor stoppen, dus terug naar hoofdmenu en toch worden we daar nog steeds over gebeld.
Foutzoeken is overigens een kunst apart, probeer eens RUN met dan een hoger regelnummer er achter, dan valt te controleren of het laatste deel van een programma bijvoorbeeld wel goed loopt. En een regel, die problemen geeft is middels een accent (hetzelfde als REM) ertussen tijdelijk uit actieve dienst te halen. Wanneer er in een PRINT commando met veel spaties en rare tekens iets niet helemaal lijkt te kloppen of het controlegetal iets afwijkt, dan is dat meestal geen ramp en blijkt na het runnen wel of iets wat fraaier op het scherm kan.

## Cursor weg.

In de vragenrubriek uit C-Info jrg.2 nummer 7 stond een manier beschreven, om de cursor van het scherm te laten verdwijnen. Hierbij was een simpel stukje Basictekst afgedrukt. Hoewel erg klein, was dit zo doeltreffend dat bij het runnen niet alleen de cursor verdween, maar de hele C-64 bleek niet meer te werken. De juiste gegevens voor regel 40 in dit programma zijn:

```
40 DATA    120,169,13,162,192,141,20,3,
142,21,3
```

Het programma werkt vervolgens naar behoren.

## Speedsave 64

In het vorige nummer hebben we in de listingrubriek abusievelijk maar een deel van de listing van speedsave afgedrukt. Er horen nog zo'n dikke 500 data-regels bij. Omdat de auteur helaas z'n listing als runtime (met machinetaal) heeft ingezonden, kunnen we op dit moment nog geen goede versie van deze uitstekende utility publiceren. Maar we hebben hem gevraagd zo snel mogelijk de complete Basic-listing van dit programma op te sturen, zodat je in het volgende nummer het programma volledig zult aantreffen.

## Brede letters.

Het programma Brede letters uit een van de vorige nummers moet op drie plaatsen worden aangepast.

```
9D(Q)=D(Q)+VAL(MID$( D$(Q),C,1)) *
  2↑X
10NEXT X: C=0 : D$(Q) = "" : NEXT Q:
  RETURN
14POKE A + W,D(1): POKE A +U, D(2):
  D(1)=0: D(2)=0: NEXT
```

Na het veranderen werkt dit programma foutloos, maar wel erg traag. De tijd die voor het berekenen van de nieuwe dubbel brede karakterset nodig is, beloopt een minuut of tien.

## Regelteller

Bij de utilities van de vorige keer stond het programma regelteller, waarmee de lengte in Bytes en het aantal Basic- regels van een programma te bepalen was. In deze listing zijn door de auteur op een paar plaatsen control-tekens gebruikt, die niet door de printer kunnen worden afgedrukt. Dit komt voor in de regels 220 en 230.
De juiste syntax van deze regels vind je hieronder.

```
220A$="(ctrl-n)(ctrl-n)!eju!qsphsbnnb!
   cfwbu!(ctrl-a)!qsphsbnnb.(ctrl-n)(ctrl-
   n)
   !sfhfmt!fo!jt!(ctrl-a)!czuft!pg!(ctrl-
   n)(ctrl-n)!nby"
230A$=A$22.001"!(ctrl-a)!ljmpc-
   zuf)t*!mboh/
   (ctrl-n)(ctrl-n)!cz!b/h/
   !wbo!efs!xbm(ctrl-n)(ctrl-a)"
```

Niet de meest logische en vooruitstrevende Basic, maar als je eenmaal de moeite hebt genomen, dit letterlijk over te typen, waarbij (ctrl-n) betekent: CTRL-toets + N-toets zal dit programma z'n werk voor je verrichten.

J.B. ●

```
17 data255,255,044,080,160,064,1
   28,128
18 data255,254,002,001,000,000,0
   00,000
19 data128,216,054,009,132,070,0
   65,032
20 data000,000,000,192,062,001,1
   28,064
21 data000,000,000,000,000,255,0
   41,013
22 data000,000,000,000,000,000,0
   01,002
23 data000,000,000,015,048,192,0
   01,002
24 data000,015,240,000,000,007,0
   56,192
25 data064,128,255,128,000,000,0
   01,002
26 data000,000,248,015,048,192,0
   00,000
27 data000,000,000,255,006,025,1
   02,152
28 data009,018,036,207,000,255,0
   03,012
29 data000,000,063,192,143,240,1
   28,064
30 data000,015,240,015,240,000,0
   00,000
31 data047,252,015,248,004,004,0
   15,244
32 data192,015,240,000,001,007,0
   15,247
33 data007,248,000,096,255,248,2
   52,252
34 data128,128,064,064,240,016,0
   16,016
35 data000,000,000,015,112,104,0
   71,092
36 data013,048,193,126,000,000,1
   92,060
37 data248,103,128,000,000,000,0
   00,000
38 data000,243,015,024,032,048,0
   15,000
39 data012,048,192,060,003,028,2
   24,128
40 data003,028,103,248,003,012,0
   48,192
41 data096,131,007,063,223,063,1
   27,126
42 data096,248,255,252,252,254,0
   62,030
43 data047,240,032,032,016,016,0
   16,016
44 data255,000,000,000,000,000,0
   00,007
45 data004,002,002,002,002,015,2
   42,002
46 data015,015,015,015,031,207,0
   15,015
47 data000,000,000,000,000,000,0
   00,000
48 data142,134,006,007,006,006,0
   06,134
49 data032,192,128,128,000,000,0
   00,000
50 data047,048,014,009,008,008,0
   07,000
51 data003,000,000,254,001,000,2
   24,031
52 data252,003,000,000,254,001,0
   00,000
53 data007,248,224,112,007,248,0
   04,004
54 data031,224,128,128,143,248,0
   00,000
55 data000,000,000,003,253,001,0
   03,003
56 data252,252,248,248,248,248,2
   52,252
57 data007,007,007,003,003,006,0
   06,014
58 data032,047,240,032,047,240,0
   00,000
59 data056,192,015,240,000,000,0
   00,000
60 data015,240,000,000,000,000,0
   00,000
61 data255,031,015,007,000,000,0
   00,000
62 data158,252,248,240,000,000,0
   00,000
63 data240,015,000,000,000,000,0
   00,000
64 data004,196,063,000,000,000,0
   00,000
65 data000,255,000,000,000,000,0
   00,000
66 data015,243,001,000,000,000,0
   00,000
67 data254,255,031,015,003,000,0
   00,000
68 data014,028,252,240,224,000,0
   00,000

*** EINDE LISTING ***
```

## Lotto

Een programma voor de gokkers. De C-64 is je behulpzaam bij het onthouden van alle lotto-cijfers, die in het verleden zijn voorgekomen. Aan de hand van die gegevens wordt dan door de 64 een prognose gemaakt voor de te verwachten uitkomst van de aanstaande lotto-trekking. Je moet uiteraard wel eerst de gegevens van het verleden intypen. Ook kun je de computer willekeurige getallen tussen 1 en 41 laten maken, zodat je op die manier geheel niet hoeft te denken, hoe jij je lotto-formulier zult moeten invullen. Het geheel is geschreven door S.G.Lageman uit Amsterdam en biedt ook nog de mogelijkheid om zowel op het scherm als op de printer een prachtig blokschema te maken van alle getallen die door de jaren heen zijn voorgekomen. Als laatste kun je ook per datum de lottocijfers teruglezen. Het programma is geschikt voor de C-64, met diskdrive en eventueel een printer.

Kortom, voor de fanatieke lotto-spelers een absolute must, en denk eraan, als je de 100.000 wint, met hulp van dit programma, laat ons dat dan even weten.....

```
  1 rem  lotto prognose / commodore-64
  2 rem  door s. lageman
  3 rem  amstrerdam / 020-1900063
100 rem ******************************
110 rem ******************************
120 rem
130 rem set up
140 rem
150 v=53280:pokev,6:pokev+1,6
160 diml(100,7),b(41),d(41),p(10):x=0
170 f1$=" loto bestand (alle jaren)":f2
    $="lotto '.. (per jaar)"
180 p$="zijn deze gegevens goed ingevul
    d[neer]"
190 b$="[RVS-aan][oranje][6xspatie]▪[RVS-ui
    t]──────────────────────────[RVS-aan]
    ▪[6xspatie]"
200 c$="[op][RVS-aan][6xspatie]"[RVS-uit]
    ▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬[RVS-aan]
    "[6xspatie][WIT]"
```

```
210 h$="[9xspatie]1[4xspatie]1[4xspatie]
    2[4xspatie]2[4xspatie]3[4xspatie]3[4xsp
    atie]4[4xspatie]4[4xspatie]5[4xspatie]
    5[4xspatie]6"
220 e$="[5xspatie]5[4xspatie]0[4xspatie]
    5[4xspatie]0[4xspatie]5[4xspatie]0[4xsp
    atie]5[4xspatie]0[4xspatie]5[4xspatie]
    0[4xspatie]5[4xspatie]0"
230 l$="l[5xspatie]l[4xspatie]l[4xspatie]
    l[4xspatie]l[4xspatie]l[4xspatie]l[4xsp
    atie]l[4xspatie]l[4xspatie]l[4xspatie]
    l[4xspatie]l[4xspatie]l"
240 rem
250 rem hoofd menu
260 rem
270 print"[CLR-HOME][HOME]"
280 printtab(6)"[oranje]UDDDDDDDDDDDDD
    DDDDDDDDDDDDDI"
290 printtab(6)"G[WIT] ** lotto ballen
    bestand **[oranje] H"
300 printtab(6)"JFFFFFFFFFFFFFFFFFFFFF
    FFFFFFK"
310 printtab(14)"[RVS-aan] menu [RVS-uit]
    [2xneer]"
320 printtab(7)"[RVS-aan] f1 [WIT][RVS-uit]
     nieuwe invoer[neer]"
330 printtab(7)"[RVS-aan][oranje] f2 [WIT]
    [RVS-uit] gegevens ophalen[3xspatie]
    (laden)[neer]"
340 printtab(7)"[RVS-aan][oranje] f3 [WIT]
    [RVS-uit] lotto overzicht[neer]"
350 printtab(7)"[RVS-aan][oranje] f5 [WIT]
    [RVS-uit] kontroleer vrij geheugen[neer
    ]"
360 printtab(7)"[RVS-aan][oranje] f6 [WIT]
    [RVS-uit] gegevens opslaan[3xspatie]
    (saven)[neer]"
370 printtab(7)"[RVS-aan][oranje] f7 [WIT]
    [RVS-uit] lotto prognose[neer]"
380 printtab(7)"[RVS-aan][oranje] f8 [WIT]
    [RVS-uit] stoppen[neer]":printtab(7
    )"=============="
390 printtab(7)" maak u-keuze ":printtab(
    7)"=============="
400 fork=0to16
410 forv=56184to56195:pokev,k:next
420 geti$:ifi$=" "ori$=""then450
430 ifasc(i$)<133or asc(i$)>140then450
440 onasc(i$)-132goto1130,490,2960,670,
    1410,450,1630,3090
450 nextk:goto400
460 rem
470 rem lotto overzicht menu
480 rem
490 gosub3180
500 printb$:print"[op][oranje][RVS-aan]
    [6xspatie][RVS-uit]l[WIT][6xspatie]
    lotto overzicht[5xspatie][oranje]l[RVS-
    aan][6xspatie]"
510 printc$
520 print"[neer][RVS-aan][oranje] f1 [RVS-u
    it][WIT] totaal overzicht"
530 print"[neer][RVS-aan][oranje] f2 [RVS-u
    it][WIT] printen totaal overzicht"
540 print"[neer][RVS-aan][oranje] f3 [RVS-u
    it][WIT] overzicht alle data"
550 print"[neer][RVS-aan][oranje] f4 [RVS-u
    it][WIT] zoeken op datum"
560 print"[neer][RVS-aan][oranje] f5 [RVS-u
    it][WIT] blok schema"
570 print"[neer][RVS-aan][oranje] f6 [RVS-u
    it][WIT] printen blok schema"
580 print"[neer][RVS-aan][oranje] f7 [WIT]
    [RVS-uit] hoofd menu"
590 print"[neer][RVS-aan][oranje] maak
    u keuze [RVS-uit][WIT]"
600 geti$:ifi$="" ori$=" "then600
610 ifasc(i$)<133orasc(i$)>139then600
620 onasc(i$)-132goto1930,2330,2630,270
    ,3700,2110,4060
630 goto600
640 rem
650 rem menu prognose
660 rem
670 gosub3180
680 printb$:print"[op][oranje][RVS-aan]
    [6xspatie][RVS-uit]l[WIT][6xspatie]
    lotto prognose[6xspatie][oranje]l[RVS-a
    an][6xspatie]":printc$
690 print"[neer][RVS-aan][oranje] f1 [RVS-u
    it][WIT] random by computer"
700 print"[neer][RVS-aan][oranje] f3 [RVS-u
    it][WIT] prognose":print"[neer][RVS-aan
    ][oranje] f7 [WIT][RVS-uit] menu"
710 geti$:ifi$=" "ori$=""then710
720 ifasc(i$)<132orasc(i$)>140then710
730 onasc(i$)-132goto770,960,710,270,71
    0,710,710,710
740 rem
750 rem random by computer
760 rem
770 k=0:t=0
780 print"[CLR-HOME][HOME]"b$:print"[op]
    [oranje][RVS-aan][6xspatie][RVS-uit]
    l[WIT][4xspatie]random by computer[4xsp
    atie][oranje]l[RVS-aan][6xspatie]":
    printc$
790 print"[6xspatie]1[4xspatie]2[4xspatie]
    3[4xspatie]4[4xspatie]5[4xspatie]6"
800 print"[oranje]====================
    =================[WIT]"
810 print"[HOME][22xneer][RVS-aan][oranje]
     f1 [RVS-uit][WIT] voor random[2xspatie
    ][RVS-aan][oranje] f7 [RVS-uit][WIT]
     menu overzicht"
820 print"[HOME][5xneer]"
830 geti$:ifi$="[ F1 ]"then860
840 ifi$="[ F7 ]"then670
850 goto830
860 k=k+1:ifk=8then770
870 forc=1to6
880 r=int((rnd(1)*41)+1):c1(c)=r:ifc>1then
    910
890 t=t+5:printtab(t)r;
900 next:t=o:print"[neer]":goto830
910 forcc=1to(c-1):ifc1(cc)=c1(c)then88
    0
920 next:goto890
930 rem
940 rem pronose by computer
950 rem
960 print"[CLR-HOME][HOME]"b$:print"[op]
    [oranje][RVS-aan][6xspatie][RVS-uit]
    l[WIT][3xspatie]prognose by compute
    r[3xspatie][oranje]l[RVS-aan][6xspatie]"
    :printc$
970 print"[op][oranje][7xspatie][RVS-aan]
    l[3xspatie]van hoog naar laag[3xspatie]
    l[WIT]"
980 print"[neer]de 10 meest gevallen ge
    tallen zijn:[neer]":k=0:gosub3480
990 fort=1to10:print"[RVS-aan]"r(t);:next
1000 print"[RVS-uit]":print"[2xneer]10 g
     etallen die het langst niet gevalle
     nzijn:[neer]":gosub3580
1010 fort=1to10:print"[RVS-aan]"w(t);:next
     :print"[RVS-uit]"
1020 print
1030 fort=1to40:print"*";:next
1040 print"[neer]een prognose kan ik nie
     t maken.":print"het beste is om van
      deze";
1050 print" getallen een[3xspatie]rijtje
      samen te stellen.[neer]"
1060 fort=1to40:print"*";:next
1070 print"druk return voor menu"
1080 geti$:ifi$<>chr$(13)then1080
1090 goto640
1100 rem
1110 rem gegevens invoer
1120 rem
1130 gosub3180
1140 printb$:print"[op][oranje][RVS-aan]
     [6xspatie][RVS-uit]l[WIT][3xspatie]
     invoer lotto gegevens[2xspatie][oranje]
     l[RVS-aan][6xspatie]":printc$
1150 print"[2xneer]"tab(16)"[oranje]-----
     --[WIT][2xop]"
1160 input"[7xrechts]datum :[RVS-aan]";d
     :print"[RVS-uit]"
1170 forb1=1to6
1180 printb1"lotto bal :[RVS-aan]";:input
     c(b1)
1190 print"[RVS-uit]":next
1200 input"reserve getal:[RVS-aan]";c(7)
     :print"[RVS-uit]"
1210 printp$:gosub3270
1220 ifi$="n"then1330
1230 rem
1240 rem b(1t/m41) aantal keren nr.
         gevallen en hoeveel weken teru
     g.
1250 rem
1260 fort=1to41:d(t)=d(t)-1:next
1270 x=1
```

```
1280 l(x,0)=d
1290 fory=1to7:l(x,y)=c(y):gosub1360:b(c
     (y))=b(c(y))+1
1300 ifb(c(y))>qthenq=b(c(y)):rem q= pro
     gnose
1310 next:i=i+1
1320 rem -x- aantal keren trekking
1330 print"[3xop][RVS-aan][oranje]** zij
     n er nog meer gegevens ** [RVS-uit]
     [WIT][neer]":gosub3270
1340 ifi$="j"then1130
1350 goto270
1360 rem d(1 t/m 41) op 0 stellen als
          getal gevallen is.
1370 d(c(y))=0:return
1380 rem
1390 rem oude gegevens ophalen
1400 rem
1410 gosub3180
1420 printb$:print"[op][oranje][RVS-aan]
     [6xspatie][RVS-uit]![WIT][2xspatie]
     oude gegevens ophalen[3xspatie][oranje]
     ![RVS-aan][6xspatie]":printc$
1430 print"[neer] keuze uit:"f1$
1440 printtab(12)f2$
1450 input"[neer] naam van file is:[RVS-aan]"
     ;f$:print"[neer] zeker weten,";
1460 gosub3270
1470 print
1480 ifi$="n"then270
1490 print"put file disk in drive and [RVS-a
     an] pres return ":print"[RVS-uit][op]"
1500 geti$:ifi$<>chr$(13)then1500
1510 open2,8,2,"0:"+f$+",s,r":gosub1830
1520 input#2,d$,i
1530 forx=0to(i-1)
1540 fory=0to7
1550 input#2,l(x,y):next:next
1560 fort=1to41:input#2,b(t),d(t):next
1570 close2:close15
1580 print"[3xneer]laatste datum van inv
     oer was "d$
1590 forw=0to3500:next:goto270
1600 rem
1610 rem gegevens opslaan
1620 rem
1630 gosub3180
1640 printb$:print"[op][oranje][RVS-aan]
     [6xspatie][RVS-uit]![WIT] gegevens
     opslaan (saven) [oranje]![RVS-aan][6xsp
     atie]":printc$
1650 print"[neer] keuze uit:"f1$
1660 printtab(12)f2$
1670 input"[neer]naam van is file :[RVS-aan]"
     ;f$:print
1680 input"[RVS-uit]geef datum[5xspatie]
     :[RVS-aan]";d$:print"[RVS-uit]"
1690 print"[neer] zeker weten, ";:gosub3
     270:ifi$="n"then270
1700 print"[2xneer]put file disk in driv
     e and [RVS-aan] pres return ":print
     "[RVS-uit]"
1710 geti$:ifi$<>chr$(13)then1710
1720 open2,8,2,"@0:"+f$+",s,w"
1730 gosub1830:k$=","
1740 print#2,d$k$i
1750 forx=0to(i-1)
1760 fory=0to7
1770 print#2,l(x,y):next:next
1780 fort=1to41:print#2,b(t)k$d(t):next
1790 close2:close15:goto270
1800 rem
1810 rem disk error reading
1820 rem
1830 open15,8,15:input#15,er,er$
1840 ifer<20thenreturn
1850 print"[RVS-aan] disk error: [RVS-uit]"
     er" "er$"[2xneer]"
1860 print"[op]druk [RVS-aan] f1 [RVS-uit]
      in voor menu"
1870 geti$:ifi$<>"[ F1 ]"then1870
1880 close15:close2:goto270
1890 rem
1900 rem
1910 rem geheel overzicht
1920 rem
```

```
1930 print"[CLR-HOME][oranje][RVS-aan][6xspa
     tie][RVS-uit]▌[WIT][2xspatie]geheel
      overzicht lotto[2xspatie][oranje]▌
     [RVS-aan][6xspatie]"
1940 printc$:print"[op] lotto[2xspatie][rech
     ts]x.[2xspatie]-weken[2xspatie]lott
     o[2xspatie][rechts]x.[2xspatie]-wek
     en"
1950 print" getal[2xspatie]gev.[2xspatie]
     terug[2xspatie]getal[2xspatie]gev.[2xsp
     atie]terug"
1960 forv=1104to1984step40
1970 pokev,117:pokev+7,93:pokev+12,93:poke
     v+19,102:pokev+27,93
1980 pokev+32,93:pokev+39,118:next
1990 forv=55376to56256step40
2000 pokev,8:pokev+7,8:pokev+12,8:pokev+
     19,8:pokev+27,8:pokev+32,8:pokev+39
     ,8
2010 next
2020 fore=1to20
2030 f=e+21:print"[rechts]"e,"[2xlinks]"
     b(e),"[6xlinks]"d(e),;
2040 print"[rechts]"f,"[2xlinks]"b(f),"[6xli
     nks]"d(f):next
2050 print"[2xrechts]21","[2xlinks]"b(21
     ),"[6xlinks]"d(21),tab(20)"[oranje]
     [RVS-aan]▌f7 menu overzicht [WIT][HOME]"
2060 geti$:ifi$<>"[ F7 ]"then2060
2070 goto490
2080 rem
2090 rem zoeken per datum
2100 rem
2110 print"[CLR-HOME][HOME]"b$:print"[op]
     [oranje][RVS-aan][6xspatie][RVS-uit]
     ▌[WIT] lotto overzicht per data [oranje
     ]▌[RVS-aan][6xspatie]":printc$
2120 print"[2xneer]"tab(24)"[oranje]----
     --[WIT][2xop]"
2130 input"geef trekkings datum: [RVS-aan]"
     ;d1:print"[RVS-uit]":ifi=0then2250
2140 forg=0to(i-1)
2150 ford2=-30000to30000step10000
2160 d3=d1+d2
2170 ifd3=l(g,0)then2200
2180 nextd2:next
2190 goto2250
2200 print"[WIT][2xrechts]datum[3xspatie]
     1[3xspatie]2[3xspatie]3[3xspatie]4[3xsp
     atie]5[3xspatie]6[3xspatie]res."
2210 print"[oranje] ===================
     ================[WIT]"
2220 gosub2560:print"[2xrechts]"j$"[rechts]"
     ;:fory=1to7
2230 printl(g,y);:gosub2450:next
2240 print:goto2260
2250 print"[2xneer]datum onbekend...."
2260 print"[HOME][14xneer][RVS-aan][oranje]
      f1 [RVS-uit][WIT] nieuwe datum [RVS-aa
     n][oranje] f7 [RVS-uit][WIT] menu o
     verzicht"
2270 geti$:ifi$="[ F1 ]"then2110
2280 ifi$="[ F7 ]"then490
2290 goto2270
2300 rem
2310 rem overzicht alle data
2320 rem
2330 m=1
2340 print"[CLR-HOME][HOME]"b$:print"[op]
     [oranje][RVS-aan][6xspatie][RVS-uit]
     ▌[WIT] overzicht lotto op datum [oranje
     ]▌[RVS-aan][6xspatie]":printc$
2350 print"[2xrechts]datum[3xspatie]1[3xspat
     ie]2[3xspatie]3[3xspatie]4[3xspatie]
     5[3xspatie]6[3xspatie]res."
2360 print"[oranje] ===================
     ================[WIT]"
2370 ifi=0then2550
2380 forg=0to(i-1):gosub2560
2390 print"[2xrechts]"j$"[rechts]";
2400 fory=1to7
2410 printl(g,y);:gosub2450:next
2420 print"[neer]":ifg/m=6then2470
2430 ifg=ithengoto2550
2440 nextg:goto2480
2450 a$=str$(l(g,y)):a=len(a$):ifa<3then
     print"[rechts]";
2460 return
2470 m=m+1
2480 print"[HOME][22xneer][RVS-aan][oranje]
      f1 [RVS-uit][WIT] voor vervolg [RVS-aa
     n][oranje] f7 [RVS-uit][WIT] menu o
     verzicht"
2490 geti$:ifi$="[ F1 ]"then2520
2500 ifi$="[ F7 ]"then490
2510 goto2490
2520 forv=1264to1864step80
2530 fort=0to39:pokev+t,32:next:next
2540 print"[HOME][5xneer]":goto2430
2550 print" geen data meer aanwezig.":goto
     2480
2560 rem lengte datum 6 pos.
2570 j$=str$(l(g,0)):j=len(j$):ifj=6then
     j$="0"+mid$(j$,2,6)
2580 ifj=7thenprint"[links]";
2590 return
2600 rem
2610 rem blok schema
2620 rem
2630 z=1829:zk=56101:gosub3180
2640 printb$:print"[op][oranje][RVS-aan]
     [6xspatie][RVS-uit]▌[WIT][4xspatie]
     blok schema lotto[5xspatie][oranje]
     ▌[RVS-aan][6xspatie]":printc$
2650 gosub3390:print"[6xneer]"tab(30)"[RVS-a
     an] "d"[links].x [RVS-uit]":print"[HOME
     ][4xneer]"
2660 fort=15to10step-1:printt"[oranje]| "
     tab(26)" |[WIT]":next
2670 fort=9to1step-1:printt" [oranje]| "tab(
     26)" |[WIT]":next
2680 print"[op][4xrechts][oranje]L______
     _______________▲"
2690 print"[3xrechts]N[rechts][WIT]12345
     67891 1 1 1 1 2":print"[2xrechts][oranj
     e]N[WIT]"tab(14)"0 2 4 6 8 0"
2700 print"[neer] [RVS-aan][oranje] f1 [RVS-
     uit][WIT] 21 t/m 41[2xspatie][RVS-aan]
     [oranje] f7 [RVS-uit][WIT] menu"
2710 forz1=1to20:ifb(z1)=0thenz=z+1
2720 ifb(z1)=0thenzk=zk+1
2730 ifb(z1)=0then2760
2740 forz2=1to(b(z1)/d):pokez-(40*(z2)),
     160:pokezk-(40*(z2)),8
2750 next:z=z+1:zk=zk+1
2760 next:goto2830
2770 forz1=21to41:ifb(z1)=0thenz=z+1
2780 ifb(z1)=0thenzk=zk+1
2790 ifb(z1)=0then2820
2800 forz2=1to(b(z1)/d):pokez-(40*(z2)),
     160:pokezk-(40*(z2)),8
2810 next:z=z+1:zk=zk+1
2820 next:goto2830
2830 geti$:ifi$="[ F1 ]"then2860
2840 ifi$="[ F7 ]"then490
2850 goto2830
2860 ifz1=21then2880
2870 ifz1=42then2630
2880 print"[4xop][5xrechts]2 2 2 2 2 3 3
      3 3 3 4":print"[5xrechts]1 3 5 7 9
      1 3 5 7 9 1"
2890 print"[neer][6xrechts] [6xrechts]20"
2900 print"[HOME][4xneer]":fort=1to14
2910 print"[5xrechts][21xspatie]":next
2920 print"[5xrechts][oranje]__________
     __________":z=1829:zk=56101:goto277
     0
2930 rem
2940 rem free memory
2950 rem
2960 gosub3180
2970 printb$:print"[op][oranje][RVS-aan]
     [6xspatie][RVS-uit]▌[WIT][7xspatie]
     free memory[8xspatie][oranje]▌[RVS-aan]
     [6xspatie]":printc$
2980 print"[2xneer]"
2990 print"u heeft nog[5xspatie]geheugen
      plaatsen vrij"
3000 forw=0to250:next
3010 r=(100-(i))*7
3020 print"[op][11xrechts]"r
3030 print"[4xneer]druk op return voor m
     enu"
3040 geti$:ifi$<>chr$(13)then3040
3050 goto270
3060 rem
3070 rem stoppen
3080 rem
3090 gosub3180
3100 printb$:print"[op][oranje][RVS-aan]
     [6xspatie][RVS-uit]▌[GEEL][2xspatie]
     ******[WIT] stoppen [GEEL]*******[2xspa
     tie][oranje]▌[RVS-aan][6xspatie]":print
     c$
3110 print"[2xneer]heeft u de gegevens a
     l opgeborgen[neer]"
```

```
3120 gosub3270
3130 ifi$="n"then3220
3140 print"[CLR-HOME][4xneer][5xrechts]t
     ot de volgende keer...."ıend
3150 rem
3160 rem scherm schoon maken
3170 rem
3180 poke56325,1:print"[CLR-HOME]":poke5
     6325,51:return
3190 rem
3200 rem pijl plaatsen
3210 rem
3220 print"[CLR-HOME]":poke1627,45:poke1
     628,62
3230 poke55899,1:poke55900,1:goto280
3240 rem
3250 rem routine ja nee
3260 rem
3270 i$="":printtab(14);:input"ja of nee
      (j/n):[2xspatie]n[3xlinks]";i$
3280 ifi$<>"n"andi$<>"j"then3330
3290 return
3300 rem
3310 rem toom voor aanslag
3320 rem
3330 fort=0to15
3340 poke54296,15:poke54277,50:poke54278
     ,248:poke54273,50:poke54272,50
3350 poke54276,33:next
3360 poke54276,0:poke54277,0:poke54278,0
3370 print"[2xop]":goto3270
3380 rem
3390 rem deler voor blok schema
3400 rem
3410 d=1:fort=1to41:n1=int((b(t)/16)+1):
     ifn1>1thend=n1
3420 ifq<b(t)thenq=b(t)
3430 next
3440 return
3450 rem
3460 rem prognose 1
3470 rem
3480 p=1:q=1
3490 fort=1to41:ifb(t)=qthen3520
3500 next:q=q-1:goto3490
3510 goto3540
3520 r(k)=t:k=k+1:ifk=11thenk=0:goto3540
3530 next
3540 return
3550 rem
3560 rem prognose 2
3570 rem
3580 q1=0:p=1:rem prognose maken
3590 fort=1to41:ifd(t)>q1thenq1=d(t)
3600 next
3610 fort=1to41:ifb(t)=q1then3640
3620 next
3630 q1=q1+1:goto3610
3640 w(k)=t:k=k+1:ifk=11then3660
3650 next:goto3630
3660 return
3670 rem
3680 rem totaal overzicht printer
3690 rem
3700 open4,4
3710 print#4,chr$(14)"[2xspatie]totaal o
     verzicht "f$chr$(15)
3720 print#4,chr$(10)
3730 print#4,"[4xspatie]lotto";
3740 print#4,chr$(16)"19x.";
3750 print#4,chr$(16)"30weken[3xspatie]="
     ;
3760 print#4,chr$(16)"44lotto";
3770 print#4,chr$(16)"59x.";
3780 print#4,chr$(16)"70weken"
3790 print#4,"[4xspatie]getal";
3800 print#4,chr$(16)"19gev.";
3810 print#4,chr$(16)"30terug[3xspatie]="
     ;
3820 print#4,chr$(16)"44getal";
3830 print#4,chr$(16)"59gev.";
3840 print#4,chr$(16)"70terug"
3850 gosub4010
3860 fore=1to20
3870 f=e+21
3880 print#4,chr$(16)"04"e;
3890 print#4,chr$(16)"19"b(e);
3900 print#4,chr$(16)"29"d(e);
3910 print#4,chr$(16)"38=";
3920 print#4,chr$(16)"44"f;
3930 print#4,chr$(16)"59"b(f);
3940 print#4,chr$(16)"69"d(f):next
3950 print#4,"[5xspatie]21";
3960 print#4,chr$(16)"19"b(21);
3970 print#4,chr$(16)"29"d(21);
3980 print#4,chr$(16)"38="
3990 gosub4010:print#4,chr$(10)
4000 print#4,chr$(16)"04laatste dag van
     invoer was: "l((x-1),0):goto4200
4010 fort=1to80:print#4,"=";:next:print#
     4:return
4020 goto4200
4030 rem
4040 rem blok schema printer
4050 rem
4060 open4,4
4070 print#4,chr$(14)"[2xspatie]block sc
     hema[2xspatie]"f$chr$(15)
4080 print#4,chr$(10)
4090 print#4,chr$(16)"10"h$
4100 print#4,chr$(16)"09"e$
4110 print#4,chr$(16)"09"l$
4120 fort=1to70:print#4,chr$(17)"¯";:next
     :print#4,chr$(145)" x."
4130 fore=1to41
4140 print#4,chr$(16)"04"e;chr$(16)"08>"
     ;:ifb(e)=0then4160
4150 forbl=0tob(e):print#4,"[RVS-aan] [RVS-u
     it]";:next:rem zwart blokje
4160 w=b(e)+2:ifb(e)=0thenw=1
4170 s$=mid$(l$,w,70)
4180 print#4,s$:next
4190 goto3990
4200 print#4:close4:goto490
```

*** EINDE LISTING ***

| | | | |
|---|---|---|---|
| regel 1 | 249 | regel 540 | 135 |
| regel 2 | 57 | regel 550 | 120 |
| regel 3 | 208 | regel 560 | 108 |
| regel 100 | 123 | regel 570 | 141 |
| regel 110 | 123 | regel 580 | 58 |
| regel 120 | 143 | regel 590 | 11 |
| regel 130 | 32 | regel 600 | 134 |
| regel 140 | 143 | regel 610 | 24 |
| regel 150 | 247 | regel 620 | 56 |
| regel 160 | 3 | regel 630 | 31 |
| regel 170 | 230 | regel 640 | 143 |
| regel 180 | 175 | regel 650 | 49 |
| regel 190 | 170 | regel 660 | 143 |
| regel 200 | 237 | regel 670 | 89 |
| regel 210 | 150 | regel 680 | 8 |
| regel 220 | 189 | regel 690 | 90 |
| regel 230 | 138 | regel 700 | 2 |
| regel 240 | 143 | regel 710 | 136 |
| regel 250 | 52 | regel 720 | 17 |
| regel 260 | 143 | regel 730 | 225 |
| regel 270 | 131 | regel 740 | 143 |
| regel 280 | 110 | regel 750 | 90 |
| regel 290 | 221 | regel 760 | 143 |
| regel 300 | 28 | regel 770 | 157 |
| regel 310 | 9 | regel 780 | 190 |
| regel 320 | 177 | regel 790 | 18 |
| regel 330 | 163 | regel 800 | 235 |
| regel 340 | 228 | regel 810 | 168 |
| regel 350 | 30 | regel 820 | 69 |
| regel 360 | 199 | regel 830 | 0 |
| regel 370 | 151 | regel 840 | 186 |
| regel 380 | 50 | regel 850 | 36 |
| regel 390 | 67 | regel 860 | 98 |
| regel 400 | 185 | regel 870 | 129 |
| regel 410 | 153 | regel 880 | 245 |
| regel 420 | 137 | regel 890 | 185 |
| regel 430 | 19 | regel 900 | 151 |
| regel 440 | 216 | regel 910 | 15 |
| regel 450 | 36 | regel 920 | 230 |
| regel 460 | 143 | regel 930 | 143 |
| regel 470 | 20 | regel 940 | 191 |
| regel 480 | 143 | regel 950 | 143 |
| regel 490 | 89 | regel 960 | 106 |
| regel 500 | 31 | regel 970 | 182 |
| regel 510 | 0 | regel 980 | 115 |
| regel 520 | 18 | regel 990 | 212 |
| regel 530 | 51 | regel 1000 | 108 |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| regel 1010 | 130 | regel 1370 | 44 | regel 1730 | 36 | regel 2090 | 189 |
| regel 1020 | 153 | regel 1380 | 143 | regel 1740 | 22 | regel 2100 | 143 |
| regel 1030 | 248 | regel 1390 | 23 | regel 1750 | 213 | regel 2110 | 68 |
| regel 1040 | 86 | regel 1400 | 143 | regel 1760 | 151 | regel 2120 | 103 |
| regel 1050 | 44 | regel 1410 | 89 | regel 1770 | 232 | regel 2130 | 18 |
| regel 1060 | 248 | regel 1420 | 145 | regel 1780 | 236 | regel 2140 | 196 |
| regel 1070 | 110 | regel 1430 | 57 | regel 1790 | 110 | regel 2150 | 120 |
| regel 1080 | 144 | regel 1440 | 100 | regel 1800 | 143 | regel 2160 | 190 |
| regel 1090 | 35 | regel 1450 | 28 | regel 1810 | 62 | regel 2170 | 95 |
| regel 1100 | 143 | regel 1460 | 89 | regel 1820 | 143 | regel 2180 | 180 |
| regel 1110 | 182 | regel 1470 | 153 | regel 1830 | 201 | regel 2190 | 82 |
| regel 1120 | 143 | regel 1480 | 124 | regel 1840 | 108 | regel 2200 | 228 |
| regel 1130 | 89 | regel 1490 | 107 | regel 1850 | 108 | regel 2210 | 247 |
| regel 1140 | 194 | regel 1500 | 141 | regel 1860 | 209 | regel 2220 | 135 |
| regel 1150 | 104 | regel 1510 | 255 | regel 1870 | 228 | regel 2230 | 139 |
| regel 1160 | 131 | regel 1520 | 191 | regel 1880 | 110 | regel 2240 | 38 |
| regel 1170 | 177 | regel 1530 | 213 | regel 1890 | 143 | regel 2250 | 120 |
| regel 1180 | 254 | regel 1540 | 151 | regel 1900 | 143 | regel 2260 | 97 |
| regel 1190 | 43 | regel 1550 | 212 | regel 1910 | 247 | regel 2270 | 38 |
| regel 1200 | 77 | regel 1560 | 149 | regel 1920 | 143 | regel 2280 | 186 |
| regel 1210 | 160 | regel 1570 | 18 | regel 1930 | 146 | regel 2290 | 84 |
| regel 1220 | 170 | regel 1580 | 164 | regel 1940 | 96 | regel 2300 | 143 |
| regel 1230 | 143 | regel 1590 | 62 | regel 1950 | 229 | regel 2310 | 133 |
| regel 1240 | 148 | regel 1600 | 143 | regel 1960 | 214 | regel 2320 | 143 |
| regel 1250 | 143 | regel 1610 | 241 | regel 1970 | 234 | regel 2330 | 48 |
| regel 1260 | 23 | regel 1620 | 143 | regel 1980 | 83 | regel 2340 | 93 |
| regel 1270 | 83 | regel 1630 | 89 | regel 1990 | 76 | regel 2350 | 223 |
| regel 1280 | 71 | regel 1640 | 57 | regel 2000 | 205 | regel 2360 | 247 |
| regel 1290 | 67 | regel 1650 | 57 | regel 2010 | 130 | regel 2370 | 41 |
| regel 1300 | 251 | regel 1660 | 100 | regel 2020 | 175 | regel 2380 | 88 |
| regel 1310 | 219 | regel 1670 | 92 | regel 2030 | 164 | regel 2390 | 33 |
| regel 1320 | 198 | regel 1680 | 133 | regel 2040 | 120 | regel 2400 | 152 |
| regel 1330 | 57 | regel 1690 | 162 | regel 2050 | 138 | regel 2410 | 139 |
| regel 1340 | 164 | regel 1700 | 252 | regel 2060 | 223 | regel 2420 | 80 |
| regel 1350 | 34 | regel 1710 | 144 | regel 2070 | 38 | regel 2430 | 201 |
| regel 1360 | 213 | regel 1720 | 177 | regel 2080 | 143 | regel 2440 | 90 |

**Moe van het overtikken?**
**Neem Uw gemak, neem een**

# INFOLIST

**cassette of diskette**

En ze werken allemaal, dus u krijgt een macht aan software voor een vriendelijke prijs. Met zo'n cassette is er in één keer al een aardig begin voor uw programma-bibliotheek en bovendien kunt u aan de hand van de gegeven programma's daar zelf weer uitbreidingen op maken.

**Beschikbaar voor de „64"**
uit no. 1 + 2 + 3 van '84 → I
uit no. 4 + 5 van '84 → II
uit no. 1 + 2 van '85 → III
uit no. 3 + 4 van '85 → IV
uit no. 5 + 6 van '85 + 'Surfmania '85' → V
uit no. 7 + 8 van '85 → VI
per diskette *f* 27,50
per cassette *f* 22,50

Wilt U bij bestelling aangeven no. I, II, III, IV, V of VI.

**Prijs *f* 22,50 incl. verzending**
ook op floppy voor *f* 27,50

## INFOLIST

## LEZERSSERVICE

*De listings uit dit en eerdere nummers van Commodore-Info zijn ook op cassette en micro-floppy te verkrijgen. Dat scheelt de lezer de tijd en ergernis van het overtypen, hetgeen tenslotte zelfs met onze listtest nog een boel werk kan betekenen, om nog maar niet te spreken van ergernis en frustratie.*
*De lezersservice van Commodore-Info wordt verzorgd door Infolist uit Arkel, Stationsweg 39B, 01831-2904, maar ook op de redaktie kunt u voor vragen meestal wel terecht 02152-63431.*

Bestellen kan alleen bij vooruitbetaling door overmaken van *f* 22,50 of *f* 27,50 op giro 3157656 Infolist te Arkel met vermelding van het infolistnummer en natuurlijk uw naam en adres. We sturen u de cassette direkt na ontvangst van uw overmaking per post op, met een kwitantie.

```
regel 2450   3
regel 2460   142
regel 2470   39
regel 2480   12
regel 2490   43
regel 2500   186
regel 2510   88
regel 2520   222
regel 2530   245
regel 2540   209
regel 2550   84
regel 2560   31
regel 2570   82
regel 2580   26
regel 2590   142
regel 2600   143
regel 2610   104
regel 2620   143
regel 2630   1
regel 2640   116
regel 2650   9
regel 2660   250
regel 2670   157
regel 2680   68
regel 2690   118
regel 2700   3
regel 2710   162
regel 2720   9
regel 2730   1
regel 2740   142
regel 2750   14
regel 2760   18
regel 2770   215
regel 2780   9
regel 2790   254
regel 2800   142
regel 2810   14
regel 2820   18
regel 2830   50
regel 2840   186
regel 2850   86
regel 2860   164
regel 2870   160
regel 2880   202
regel 2890   172
regel 2900   47
regel 2910   42
regel 2920   133
regel 2930   143
regel 2940   138
regel 2950   143
regel 2960   89
regel 2970   4
regel 2980   255
regel 2990   93
regel 3000   177
regel 3010   14
regel 3020   255
regel 3030   81
regel 3040   142
regel 3050   34
regel 3060   143
regel 3070   184
regel 3080   143
regel 3090   89
regel 3100   144
regel 3110   35
regel 3120   89
regel 3130   170
regel 3140   178
regel 3150   143
regel 3160   135
regel 3170   143
regel 3180   211
regel 3190   143
regel 3200   22
```

```
regel 3210   143
regel 3220   220
regel 3230   150
regel 3240   143
regel 3250   24
regel 3260   143
regel 3270   188
regel 3280   108
regel 3290   142
regel 3300   143
regel 3310   11
regel 3320   143
regel 3330   193
regel 3340   16
regel 3350   237
regel 3360   104
regel 3370   142
regel 3380   143
regel 3390   26
regel 3400   143
regel 3410   252
regel 3420   7
regel 3430   130
regel 3440   142
regel 3450   143
regel 3460   45
regel 3470   143
regel 3480   185
regel 3490   225
regel 3500   127
regel 3510   85
regel 3520   216
regel 3530   130
regel 3540   142
regel 3550   143
regel 3560   46
regel 3570   143
regel 3580   115
regel 3590   102
regel 3600   130
regel 3610   21
regel 3620   130
regel 3630   30
regel 3640   240
regel 3650   17
regel 3660   142
regel 3670   143
regel 3680   54
regel 3690   143
regel 3700   51
regel 3710   36
regel 3720   113
regel 3730   9
regel 3740   230
regel 3750   16
regel 3760   240
regel 3770   234
regel 3780   156
regel 3790   228
regel 3800   112
regel 3810   29
regel 3820   203
regel 3830   116
regel 3840   169
regel 3850   82
regel 3860   175
regel 3870   74
regel 3880   159
regel 3890   56
regel 3900   59
regel 3910   158
regel 3920   164
regel 3930   61
regel 3940   193
regel 3950   218
regel 3960   86
```

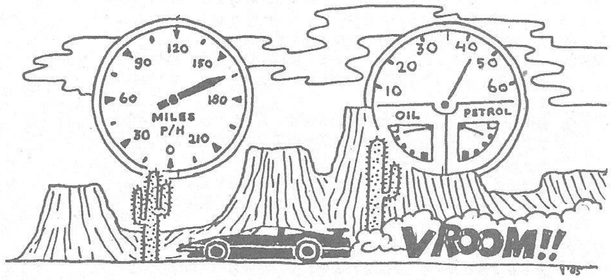
```
regel 3970   89
regel 3980   99
regel 3990   253
regel 4000   221
regel 4010   60
regel 4020   79
regel 4030   143
regel 4040   140
regel 4050   143
regel 4060   51
regel 4070   189
regel 4080   113
regel 4090   136
```

```
regel 4100   141
regel 4110   148
regel 4120   21
regel 4130   178
regel 4140   52
regel 4150   150
regel 4160   29
regel 4170   202
regel 4180   43
regel 4190   94
regel 4200   58

ready.
```

## Woestijnrace 64

Een prachtig spel in verbazend weinig regels. Deze prijsvraaginzending doet precies wat we hadden gehoopt, namelijk een optimaal resultaat, zonder dat het overtypen nu direkt een week kost. Een creatief gebruik van sprites en een goede uitvoering van het autoraceprincipe. Men moet op een weg andere auto's inhalen en bij botsingen verschijnt er een "fraai" doodskopje. Compleet met wijzers en meters, dit is werkelijk een van de fraaiste programma's van deze lengte tot nu toe.




```
  1 rem woestijnrace / cbm-64
  2 rem door a reitsma
  3 rem heereveen / 05130-24759
  4 rem
 10 poke56,32:clr:print"[CLR-HOME]
    [2xneer][ROOD]"tab(14)"woesti
    jnrace[neer]"tab(55)"[ZWART]d
    oor a.X.r":poke918,5
 20 v=53248:s=54272:pokev+32,0:poke
    v+33,7:f=255:pokev+21,f:pokev
    +27,0:pokev+28,f
 30 forx=0to7:a=(x>4):pokev+39+x,
    6+a:poke2040+x,129-a:next:poke
    v+37,9:pokev+38,0
 40 pokev+39,1:forx=stos+24:pokex
    ,0:next:pokes+22,30:pokes+23,
    241:pokes+24,31
 50 pokes+12,28:pokes+8,9:pokes+1
    3,9:pokes+5,33:pokes+15,80:poke
    s+19,9:pokes+20,5
 60 print"[2xneer]"chr$(8)chr$(14
    2)tab(13)"40 sec. geduld":a=4
    :p=8256:forx=1to9:reada$
 70 fory=1to67:h=asc(mid$(a$,y)):
    g=asc(mid$(a$,y+1)):h=h-35+96
    *(h>191):a=a-(a=0)*4
 80 g=g-35+96*(g>191):pokep,(h*aand
    255)+g*a/64:a=a*4and255:y=y-(
    a=0):p=p+1:nexty,x
 90 a$="[RVS-aan][GEEL][8xspatie]
    [grijs 2][18xspatie][GEEL][3xspat
    ie][ZWART][3xspatie][BLAUW]":
    b$=a$+"UCCCCCCI":c$=a$+"JCCCC
    CCK"
100 d$=a$+"B[6xspatie]B":printb$c
    $b$c$b$a$"BpuntenB"d$c$b$c$b$
    a$"[BLAUW]Bauto'sB"d$c$b$c$;
```

```
110 printb$a$"BlevensB"d$c$b$c$b$
    c$a$"[7xspatie][links][INSert]
     [grijs 2]";:s$="[6xspatie]":
    sys8448:goto270
120 r=q*10:sys8662"[RVS-aan]"r;:sys
    8666t;:sys8651w;:forx=1to7:poke
    900+x,0:poke908+x,0
130 pokev+x*2,0:pokev+1+x*2,f/(u+
    1)*xandf:p(x)=f:next:sys8670"
    [RVS-aan][grijs 2]druk vuurkn
    op";
140 pokev,150:pokev+1,160:pokes+1
    ,3:poke919,3:poke916,1:pokev+
    30,0:pokev+21,f
150 poke2040,129:wait56320,16,16:
    sys8670s$s$" [BLAUW]";:poke91
    7,1:ti$="000000"
160 fory=5to7:ifpeek(v+1+y*2)>0then
    180
170 pokev+1+y*2,20:p=rnd(0)*50:poke
    v+y*2,14-p*(p>10)-(p<=10)*218
180 w=w+peek(916)/10:forx=5-uto4:
    a=peek(v+1+x*2):ifa>0then210
190 pokev+x*2,91+rnd(0)*111:poke9
    00+x,rnd(0)*5-2and255:p=rnd(0
    )*3+u/2
200 poke908+x,255-p:p(x)=20-(p>peek
    (916))*229:pokev+1+x*2,p(x):goto
    230
210 ifp(x)<150anda>=150thenw=w+25
    :r=r-1:sys8662r"[links] ";:poke
    s+18,0:goto280
220 ifp(x)<150thenp(x)=a
230 sys8651int(w);:ifnotpeek(v+30
    )and1andti<5000thennextx,y:goto
    160
240 poke917,0:poke2040,131:poke54
    276+7,0:poke54276+7,129:forx=
    1to9*f:next:a=t>0
250 w=int(w):t=t+a:sys8666t;:on-a
    goto120:pokev+21,224:pokev,0:
    ifw>zthenz=w
260 sys8670"[grijs 2] einde spel"
    ;:sys8670"[2xneer] rekord:"z;
    :wait56320,16,16:sys8670s$s$;
270 sys8670"[2xneer][3xspatie]"s$
    s$"[BLAUW]";:sys8651s$;:sys86
    66s$;:w=0:t=3:u=1:q=1:goto120
280 pokes+18,17:on-(r>0)goto220:poke
    917,0:n=int(ti/50):q=q+1:u=u-
    (u<4)
290 forx=1to4:pokev+1+x*2,0:next:
    b=1000-n*20:b=-b*(b>0):w=int(
    w+b):t=t+1
300 sys8670"[op][grijs 2]tijd:"n"
    (s)";:sys8670"[neer]bonus:"b;
310 forx=1to9*f:next:sys8670"[grijs 2
    ][op]"s$s$;:sys8670"[neer]"s$
    s$"[BLAUW]";:goto120
320 data"#mm#/.[/B-[BB.[BB↑nBB.[B
    B-kB/-k/#mm#%]nc%rAc%]nc%↑↑c%
    ↑↑c1↑↑oB*7BB(wB
330 data"Bz:BB(wBB*7B/##/#&s##1o#
    B1←&nAo&qAo&mBo&mAo1qA←1mAn]↑
    1]qn&nmo#Ans#&m
340 data"s#&qs#&]s#&ms############
    ############%mc#-mk#mmm#nm]#r↑
    A#r.u#r↑A#nm]#-
350 data"mk#-qk#%Bc#%mc##m##cm%%k
    #-e]enf2]←u-qkenv]fo#1ss#&&Bm
    4q0'@%1b[w0z-g=
360 data"f43&m5)0(311c;w=w)%1#;w<
    w-j]f4.3n<7&w#0//ame$]#1q3*3←
    &+;><3&)*y/#←j]
370 data"o#6,%.#%m3%<#@%£#0#;b;3&
    i3&3u8q3$/j2g#q-d#0,Btd1#<y'#
    [e+up&'n3&?5b--
380 data"v1;&w%]e%+]9#£#4n13&[#es
    %qe1h#1←#%-/hs0-g$5qh#2c#1#0u
    ns#+ku:#[]7#t#$
390 data"5m[#w'm3$q$=g#.-ugm3$q&3
    o#.k]+[#w-g#f367m5)0$04/£%)e$
    m#d)%&sBwucmm+/
400 data"w21e'p&rkcmc%p&nn,c&v,?&
    ←#u3#[f3#?e/i#1/#@4//ak###&BB
    sk&$#7*%3s3(###

*** EINDE LISTING ***


woestijnrace


regel 1    181          regel 200   156
regel 2    25           regel 210   70
regel 3    134          regel 220   96
regel 4    143          regel 230   241
regel 10    101         regel 240   103
regel 20    52          regel 250   165
regel 30    30          regel 260   148
regel 40    59          regel 270   67
regel 50    26          regel 280   156
regel 60    255         regel 290   186
regel 70    238         regel 300   246
regel 80    101         regel 310   110
regel 90    146         regel 320   129
regel 100    163        regel 330   116
regel 110    194        regel 340   22
regel 120    219        regel 350   50
regel 130    33         regel 360   95
regel 140    169        regel 370   2
regel 150    48         regel 380   143
regel 160    167        regel 390   178
regel 170    12         regel 400   118
regel 180    15
regel 190    154        ready.
```

## Spelrecord 64

Wie veel spelletjes met zijn computer speelt, en daarbij hoger scoort dan zijn vriendjes, wil misschien best bijhouden wie er nu wel de hoogste score heeft behaald. Met dit programma valt dat bij te houden. Het is in wezen een wat aangepast database-programma en met wat moeite is het wel aan te passen voor andere toepassingen. Met name de schermopbouw en de invoerroutines zijn hier relatief eenvoudig te volgen en dus aan te passen, zoals in regel 310-330 of 420-440, waar de naam als input a$ en sp$ wordt gevraagd en even getest wordt hoe lang dat dan wel niet is, en eventueel wordt ingekort. Dit soort dingen vragen wel wat programmeerregels, maar maken een programma in het gebruik wel vriendelijker. We proberen in Commodore-Info vaak zo veel mogelijk korte programma's te plaatsen, waarin dit soort eenvoudige grapjes niet voorkomen, maar dat betekent niet dat men er zelf niet mee kan werken of een ander programma wat kan verbeteren. Het is alleen zo, dat wie eenmaal wat verder is met programmeren, die dingen wel weet en automatisch toevoegt bij het afwerken van het programma.

```
 1 rem  spelrecord/cbm-64
 2 rem  door gert nentjes
 3 rem  uit rotterdam.
 4 rem
50 dima$(10,10):poke53280,1:poke
   53281,1:poke646,2:printchr$(1
   4)
60 rem   poke788,52:rem run/stop
    uitschakelen
70 bl$="[40xspatie]"
80 b$="Bijhouden puntenstanden v
   an spellen":gosub1680
90 print"[2xneer]De puntenstand
   wordt gesorteerd getoond[2xneer]"
```

```
100 print"Maximaal 9 spellen en 9
     spelers[2xneer]"
110 print"Bestandsopslag op casse
    tte of disk[2xneer]"
120 print"Geluid bij bijzondere s
    ituaties[neer]":gosub1570
130 b$="Bestaan er al gegevens ?[2xsp
    atie](j/n)":gosub1680
140 getp$:on-(p$="")goto140:ifp$=
    "j"thengosub1230
150 b$="***[3xspatie]menu[3xspatie]
    ***":gosub1680
160 print"[neer][5xspatie]1. invo
    eren nieuw spel"
170 print"[neer][5xspatie]2. invo
    eren nieuwe speler"
180 print"[neer][5xspatie]3. invo
    eren nieuwe punten"
190 print"[neer][5xspatie]4. over
    zicht per spel"
200 print"[neer][5xspatie]5. over
    zicht per speler"
210 print"[neer][5xspatie]6. opsl
    aan gegevens"
220 print"[neer][5xspatie]7. vera
    nderen naam spel"
230 print"[neer][5xspatie]8. vera
    nderen naam speler"
240 gosub1600
250 getp$:on-(p$="")goto250:onval
    (p$)goto280,380,660,770,920,1
    380,1010,1100
260 ifval(p$)<1orval(p$)>8then250
270 rem *** invoeren spellen ***
280 b$="Invoeren nieuw spel":gosub
    1680
290 ifx<10thenvv$="het spel":gosub
    480:x=x+1
300 ifx>9thengosub1740:print"[2xneer]
    er zijn 9 spellen ingevoerd[2xspa
    tie](=maximum)":gosub1660:goto
    150
310 print"[neer]Naam spel "x:input
    a$(x,1)
320 ifa$(x,1)=chr$(42)thena$(x,1)
    =left$(bl$,15):x=x-1:goto150
330 iflen(a$(x,1))>15thengosub180
    0:print"[neer]naam wordt afge
    kort":a$(x,1)=left$(a$(x,1),1
    5)
340 a$(x,1)=a$(x,1)+left$(bl$,15-
    len(a$(x,1)))
350 forp=1toy:a$(x,p)=left$(a$(x,
    1),15)+mid$(a$(1,p),16,15)+left$
    (bl$,10):nextp
360 x=x+1:goto300
370 rem *** invoeren spelers ***
380 b$="Invoeren nieuwe speler":gosub
    1680
390 if x=0thengosub1620:gosub1660
    :goto150
400 ify<10then vv$="de speler":gosub
    480:y=y+1
410 ify>9thengosub1740:print"[2xneer]
    er zijn 9 spelers ingevoerd (
    =maximum)":gosub1660:goto150
420 print"[neer]Naam speler "y:input
    sp$(y)
430 ifsp$(y)=chr$(42)thensp$(y)=left$
    (bl$,15):y=y-1:goto150
440 iflen(sp$(y))>15thengosub1800
    :print"[neer]naam wordt afgek
    ort":sp$(y)=left$(sp$(y),15)
450 sp$(y)=sp$(y)+left$(bl$,15-len
    (sp$(y)))
460 forp=1to9:a$(p,y)=left$(a$(p,
    1),15)+sp$(y)+left$(bl$,10):next
    p
470 y=y+1:goto410
480 print"[2xneer]Geef de [RVS-aan]
     naam [RVS-uit] van "vv$
490 print"[neer]maximaal 15 lette
    rs"
500 print"[neer]beeindig door : *
    [2xneer]":return
510 if x=0thengosub1620:gosub1660
    :goto150
520 printtab(11)"[RVS-aan]kies he
    t spel[RVS-uit][2xneer]"
530 max=x:ifx>9thenforp=1tomax-1
540 ifx<10thenforp=1tomax
550 print p spc(2)left$(a$(p,1),1
    5):nextp
560 getp$:on-(p$="")goto560:x1=val
    (p$):ifx1<1orx1>max then560
570 return
580 ify=0thengosub1800:print"Nog
    geen speler ingevoerd, doe da
    t eerst":gosub1660:goto150
590 printtab(11)"[RVS-aan]kies de
     speler[RVS-uit][2xneer][3xspatie
    ]"
600 max=y:ify>9thenforp=1tomax-1
610 ify<10thenforp=1tomax
620 print p spc(2)mid$(a$(1,p),16
    ,15):nextp
630 getp$:on-(p$="")goto630:y1=val
    (p$):ify1<1ory1>max then630
640 return
650 rem *** invoeren punten ***
660 ifx=.thenify=.thengosub1620:gosub
    1660:goto150
670 b$="Invoeren nieuwe punten":gosub
    1680:gosub510:gosub1680:gosub
    580
680 print"[2xneer][RVS-aan] "mid$
    (a$(x1,y1),16,15)" [RVS-uit][2xne
    er]"
690 input"punten :";pp$
700 pp$=pp$+left$(bl$,10-len(pp$)
    )
710 ifval(right$(a$(x1,y1),10))<val
    (pp$)then750
720 gosub1800:print"[neer][RVS-aan]
    Huidige puntenstand is :[RVS-uit]
     "right$(a$(x1,y1),10)
730 print"[neer]moet dat verander
    d worden ? (j/n)
740 getp$:on-(p$="")goto740:ifnot
    p$="j"then150
750 a$(x1,y1)=left$(a$(x1,y1),30)
    +pp$:goto800
760 rem *** overzicht spellen ***
770 b$="Overzicht per spel":gosub
    1680
780 ifx=0thengosub1640:gosub1660:
    goto150
790 gosub510
800 b$=left$(a$(x1,1),15):gosub16
    80:rem regel 800 is sprongadr
    es
810 rem *** sorteren ***
820 max=y:forp=1tomax:d$(p)=a$(x1
    ,p):nextp
830 fork=1tomax:forj=ktomax
840 ifval(right$(d$(k),10))<val(right$
```

```
     (d$(j),10))thenc$=d$(j):d$(j)
     =d$(k):d$(k)=c$
850 nextj:nextk
860 fork=1tomax
870 ifval(right$(d$(k),10))<>0then
     printmid$(d$(k),16,15)spc(2)right$
     (d$(k),10):print
880 nextk
890 q=0:forp=1tomax:q=q+val(right$
     (d$(p),10)):nextp:ifq=0thengosub
     1640:gosub1660:goto150
900 gosub1570:goto150
910 rem *** overzicht spelers ***
920 b$="Overzicht per speler":gosub
     1680
930 ifx=0ory=0thengosub1640:gosub
     1660:goto150
940 gosub580:b$=mid$(a$(1,y1),16,
     15):gosub1680
950 max=x:forp=1tomax
960 ifnotval(right$(a$(p,y1),10))
     =0thenprintleft$(a$(p,y1),15)
     spc(2)right$(a$(p,y1),10):print
970 nextp
980 q=0:forp=1tomax:q=q+val(right$
     (a$(p,y1),10)):nextp:ifq=0then
     gosub1640:gosub1660:goto150
990 gosub1570:goto150
1000 rem *** veranderen spelnaam *
     **
1010 b$="Veranderen spelnaam":gosub
     1680
1020 vv$="spelnaam":gosub1180
1030 ifnotp$="j"then150
1040 print"[neer]":gosub510
1050 print"[neer]spelnaam (max. 15
      letters)":inputsp$(x1)
1060 iflen(sp$(x1))>15thengosub180
     0:print"[neer]Afkorting naam"
     :sp$(x1)=left$(sp$(x1),15)
1070 sp$(x1)=sp$(x1)+left$(bl$,15-
     len(sp$(x1)))
1080 forp=1toy:a$(x1,p)=sp$(x1)+mid$
     (a$(x1,p),16,15)+left$(bl$,10
     ):nextp:goto150
1090 rem *** veranderen spelernaam
     ***
1100 b$="Veranderen naam speler":gosub
     1680
1110 vv$="spelernaam":gosub1180
1120 ifnotp$="j"then150
1130 print"[neer]":gosub580
1140 print"[neer]spelernaam (max.
     15 letters)":inputsp$(y1)
1150 iflen(sp$(y1))>15thengosub180
     0:print"[neer]Afkorting naam
     !":sp$(y1)=left$(sp$(y1),15)
1160 forp=1tox:a$(p,y1)=left$(a$(p
     ,y1),15)+sp$(y1)+left$(bl$,10

1170 goto150
1180 gosub1800:print"[2xneer]Een "
     vv$" kan veranderd worden.[neer]"
1190 print"Gegevens van de verande
     rde "vv$:print"[neer]worden g
     ewist.[3xneer]"
1200 print"Veranderen ?[3xspatie][RVS-
     aan](j/n)[RVS-uit]"
1210 getp$:on-(p$="")goto1210:return
1220 rem *** laden data  ***
1230 b$="Gegevens staan op :":gosub
     1680
1240 print"[2xneer][12xspatie]1. c
     assette"
1250 print"[2xneer][12xspatie]2. d
     isk"
1260 print"[4xneer][12xspatie]3. t
     erug menu"
1270 gosub1600
1280 getp$:on-(p$="")goto1280:onval
     (p$)goto1300,1310,150
1290 ifval(p$)<1orval(p$)>3then128
     0
1300 open2,1,0,"spel-punten":goto1
     320
1310 open1,8,1:open2,8,2,"0:spel-p
     unten,s,r"
1320 input#2,x:input#2,y
1330 fora=1tox:forb=1toy
1340 input#2,a$(a,b)
1350 nextb:nexta
1360 close2:close1:return
1370 rem *** saven data  ***
1380 b$="Opslaan op:":gosub1680
1390 print"[2xneer][12xspatie]1. c
     assette"
1400 print"[2xneer][12xspatie]2. d
     isk"
1410 print"[4xneer][12xspatie]3. t
     erug menu"
1420 gosub1600
1430 getp$:on-(p$="")goto1430:onval
     (p$)goto1450,1460,150
1440 ifval(p$)<1orval(p$)>3then143
     0
1450 open2,1,1,"spel-punten":goto1
     470
1460 open1,8,1:open2,8,2,"@0:spel-
     punten,s,w"
1470 print#2,x:print#2,y
1480 fora=1tox:forb=1toy
1490 print#2,a$(a,b)
1500 nextb:nexta
1510 close2:close1
1520 print"[4xneer]Wil je stoppen
     ? [RVS-aan](j/n)[RVS-uit]"
1530 getp$:on-(p$="")goto1530:ifnot
     p$="j"then150
1540 print"[CLR-HOME][5xneer]EINDE
      programma, tot ziens":end
1550 :
1560 rem *** subroutines ***
1570 poke211,7:poke214,23:sys58732
     :print"[RVS-aan][4xspatie]dru
     k op een toets[4xspatie][RVS-uit]"
1580 poke198,0:wait198,1:poke198,0
     :return
1590 :
1600 print"[3xneer][RVS-aan][13xspatie
     ]maak uw keuze[13xspatie][RVS-uit
     ]":return
1610 :
1620 gosub1800:print"Nog geen spel
      ingevoerd, doe dat eerst !":
     return
1630 :
1640 gosub1740:print"[2xneer][RVS-aan]
     [8xspatie]Nog geen gegevens b
     ekend[8xspatie][RVS-uit]":return
1650 :
1660 forp=1to1500:nextp:return
1670 :
1680 p=(40-len(b$))/2
1690 print"[CLR-HOME][RVS-aan]"bl$
1700 print"[RVS-aan][op]"left$(bl$
```

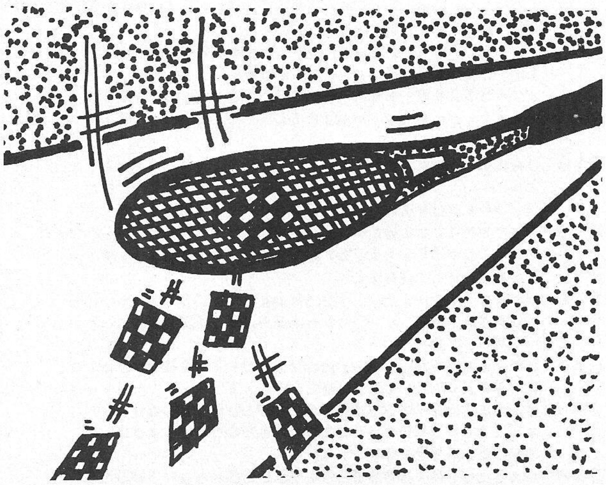
```
     ,p);b$;left$(bl$,p+1)
1710 print"[RVS-aan][op]"bl$"[RVS-uit]"
     :return
1720 rem *** geluid ***
1730 :
1740 forra=1to3
1750 forz=1to10:poke54272+z,0:next
     :poke54277,129 :poke54278,129
     :poke54296,15
1760 forz=1to50step10:poke54273,z+
     25:poke54276,17
1770 poke54272,z+10:nextz:poke5427
     2,0:poke54273,0
1780 nextra:return
1790 :
1800 forra=1to3
1810 forz=1to10:poke54272+z,0:next
     :poke54277,96 :poke54278,0:poke
     54296,15
1820 forz=50to10step-10:poke54273,
     z+5:poke54276,33
1830 poke54272,z+1:nextz:poke54272
     ,0:poke54273,0
1840 nextra:return
```

spelrecord

| | | |
|---|---|---|
| regel | 1 | 26 |
| regel | 2 | 12 |
| regel | 3 | 97 |
| regel | 4 | 143 |
| regel | 50 | 170 |
| regel | 60 | 0 |
| regel | 70 | 168 |
| regel | 80 | 233 |
| regel | 90 | 18 |
| regel | 100 | 255 |
| regel | 110 | 84 |
| regel | 120 | 56 |
| regel | 130 | 63 |
| regel | 140 | 215 |
| regel | 150 | 35 |
| regel | 160 | 111 |
| regel | 170 | 76 |
| regel | 180 | 92 |
| regel | 190 | 41 |
| regel | 200 | 193 |
| regel | 210 | 180 |
| regel | 220 | 142 |
| regel | 230 | 38 |
| regel | 240 | 84 |
| regel | 250 | 210 |
| regel | 260 | 90 |
| regel | 270 | 4 |
| regel | 280 | 148 |
| regel | 290 | 83 |
| regel | 300 | 96 |
| regel | 310 | 65 |
| regel | 320 | 24 |
| regel | 330 | 179 |
| regel | 340 | 185 |
| regel | 350 | 152 |
| regel | 360 | 147 |
| regel | 370 | 15 |
| regel | 380 | 112 |
| regel | 390 | 175 |
| regel | 400 | 149 |
| regel | 410 | 108 |
| regel | 420 | 223 |
| regel | 430 | 38 |
| regel | 440 | 197 |
| regel | 450 | 203 |
| regel | 460 | 63 |
| regel | 470 | 151 |
| regel | 480 | 149 |
| regel | 490 | 193 |
| regel | 500 | 167 |
| regel | 510 | 175 |
| regel | 520 | 18 |
| regel | 530 | 184 |
| regel | 540 | 220 |
| regel | 550 | 4 |
| regel | 560 | 173 |
| regel | 570 | 142 |
| regel | 580 | 24 |
| regel | 590 | 81 |
| regel | 600 | 186 |
| regel | 610 | 221 |
| regel | 620 | 153 |
| regel | 630 | 172 |
| regel | 640 | 142 |
| regel | 650 | 203 |
| regel | 660 | 24 |
| regel | 670 | 214 |
| regel | 680 | 62 |
| regel | 690 | 220 |
| regel | 700 | 191 |
| regel | 710 | 13 |
| regel | 720 | 103 |
| regel | 730 | 155 |
| regel | 740 | 200 |
| regel | 750 | 13 |
| regel | 760 | 92 |
| regel | 770 | 75 |
| regel | 780 | 177 |
| regel | 790 | 35 |
| regel | 800 | 169 |
| regel | 810 | 253 |
| regel | 820 | 37 |
| regel | 830 | 197 |
| regel | 840 | 204 |
| regel | 850 | 211 |
| regel | 860 | 57 |
| regel | 870 | 227 |
| regel | 910 | 103 |
| regel | 920 | 226 |
| regel | 930 | 156 |
| regel | 940 | 239 |
| regel | 950 | 104 |
| regel | 960 | 108 |
| regel | 970 | 210 |
| regel | 980 | 76 |
| regel | 990 | 179 |
| regel | 1000 | 198 |
| regel | 1010 | 173 |
| regel | 1020 | 168 |
| regel | 1030 | 36 |
| regel | 1040 | 75 |
| regel | 1050 | 141 |
| regel | 1060 | 151 |
| regel | 1070 | 91 |
| regel | 1080 | 5 |
| regel | 1090 | 93 |
| regel | 1100 | 68 |
| regel | 1110 | 63 |
| regel | 1120 | 36 |
| regel | 1130 | 82 |
| regel | 1140 | 37 |
| regel | 1150 | 187 |
| regel | 1160 | 25 |
| regel | 1170 | 31 |
| regel | 1180 | 126 |
| regel | 1190 | 250 |
| regel | 1200 | 66 |
| regel | 1210 | 91 |
| regel | 1220 | 9 |
| regel | 1230 | 22 |
| regel | 1240 | 186 |
| regel | 1250 | 138 |
| regel | 1260 | 62 |
| regel | 1270 | 84 |
| regel | 1280 | 239 |
| regel | 1290 | 137 |
| regel | 1300 | 190 |
| regel | 1310 | 112 |
| regel | 1320 | 175 |
| regel | 1330 | 126 |
| regel | 1340 | 71 |
| regel | 1350 | 193 |
| regel | 1360 | 165 |
| regel | 1370 | 34 |
| regel | 1380 | 89 |
| regel | 1390 | 186 |
| regel | 1400 | 138 |
| regel | 1410 | 62 |
| regel | 1420 | 84 |
| regel | 1430 | 248 |
| regel | 1440 | 134 |
| regel | 1450 | 197 |
| regel | 1460 | 181 |
| regel | 1470 | 215 |
| regel | 1480 | 126 |
| regel | 1490 | 91 |
| regel | 1500 | 193 |
| regel | 1510 | 221 |
| regel | 1520 | 64 |
| regel | 1530 | 246 |
| regel | 1540 | 182 |
| regel | 1550 | 58 |
| regel | 1560 | 238 |
| regel | 1570 | 95 |
| regel | 1580 | 247 |
| regel | 1590 | 58 |
| regel | 1600 | 198 |
| regel | 1610 | 58 |
| regel | 1620 | 16 |
| regel | 1630 | 58 |
| regel | 1640 | 126 |
| regel | 1650 | 58 |
| regel | 1660 | 242 |
| regel | 1670 | 58 |
| regel | 1680 | 187 |
| regel | 1690 | 52 |
| regel | 1700 | 197 |
| regel | 1710 | 208 |
| regel | 1720 | 69 |
| regel | 1730 | 58 |
| regel | 1740 | 206 |
| regel | 1750 | 102 |
| regel | 1760 | 171 |
| regel | 1770 | 165 |
| regel | 1780 | 221 |
| regel | 1790 | 58 |
| regel | 1800 | 206 |
| regel | 1810 | 205 |
| regel | 1820 | 82 |
| regel | 1830 | 117 |
| regel | 1840 | 221 |

## Crazy Sprite 64

Een kort actiespelletjes voor twee spelers met joysticks, waarbij de bal in de vorm van een vreemd figuurtje, "de crazy sprite" ook nog al gestoord kan reageren. Een vrij kort, maar leuk spel met nogal veel machinetaal, dat via peek en pokes en dataregels wordt weggezet.



```
1 rem crazy sprite / cbm-64
2 rem door jg vd bosh
3 rem utrecht / 030-946160
4 rem
```

```
 20 print"[CLR-HOME][WIT][6xneer]"
    tab(15)"30 sec aub..[neer]":f
    $="[CLR-HOME][6xneer]fout in
    data,regel":px=49399
 30 fori=1to10:reada:next:gosub16
    0:sys49152:gosub200:py=px+1:s
    1=py+1:s2=s1+1
 40 print"[4xneer][WIT]"tab(16)"1
     langzaam":printtab(16)"2 sne
    l":printtab(16)"3 gestoord"
 50 printtab(16)"4 gek":printtab(
    16)"5 krankzinnig":printtab(1
    6)"[neer]kies een spel"
 60 geta$:ifa$<"1"ora$>"5"then60:
    data4,2,8,4,4,-4,7,-5,10,-6
 70 a=val(a$):restore:ifa>1thenfor
    i=1to2*(a-1):readd:next
 80 print"[CLR-HOME]":fori=0to10:
    print"[HOME][12xneer]"tab(19)
    10-i"[links] ":forj=1to300:next
    :next
 90 readdx,dy:pokepy,abs(dy):poke
    s1,0:pokes2,0:print"[CLR-HOME]"
    :f1=0:f2=0
100 a=(abs(-256*(n>127)-dx))and25
    5:pokepx,a:q$="[23xneer]"
110 if dy<0 then p=(peek(py)-1+3*
    rnd(0))and255:ifp<-dyorp>256+
    dythenpokepy,p
120 print"[HOME][ZWART]punten spe
    ler 1:"f2:printq$;:print"punt
    en speler 2:"f1;
130 ifabs(f2-f1)<2or(f1<15andf2<1
    5)thenf1=peek(s1):f2=peek(s2)
    :goto110
140 n=peek(px):pokepx,0:pokepy,0:
    print"[HOME][6xneer][grijs 3]"
    tab(10)"gefeliciteerd speler"
    ;
150 fori=1to10:getd$:next:a=1-(f2
    <f1):printa:printtab(11)"je h
    ebt gewonnen!!!":goto40
160 fori=49152to49631step32:readd
    $,checksum:c=int(i/256):forj=
    0tolen(d$)-2step2
170 h=asc(mid$(d$,j+1))-48:h=h+7*
    (h>9):l=asc(mid$(d$,j+2))-48:
    l=l+7*(l>9)
180 c=(c+l+2*h)and255:pokei+j/2,1
    6*h+l:next:ifch=candlen(d$)=6
    4thennext:return
190 printf$":[neer]":printi/32-15
    12"data"d$","ch:end
200 ad=53248:for i=192*64 to 193*
    64-1:pokei,-63*(int((i+1)/3)=
    (i+1)/3)
210 next:poke2040,192:poke2041,19
    2:            pokead+21,11:pokea
    d,30:pokead+2,40
220 fori=0to10:forj=0to2:reada: poke
    12416+3*i+j,a:poke12476-3*i+j
    ,a:next:next
230 poke ad+16,2:pokead+1,100:poke
    ad+3,100 : pokead+39,12:pokea
    d+40,12
240 pokead+6,100:pokead+7,100:poke
    ad+42,15 :poke2043,194 : return
250 data78a9018d0ddca91b8d11d0a96
    a8d1403a9c08d1503a9018d1ad08d
    19d0ad19d0, 70
260 datac901d0038d19d058eaa930853
    4853878a50129fb8501a200a008bd
    00d09d0038, 246
270 datae8d0f7ee3cc0ee3fc088d0eea
    9d08d3cc0a9388d3fc0a501090485
    0158eaad18, 189
280 datad029f0090e8d18d060ea2082c
    0a9018d19d0a9518d12d0ad00c1d0
    034c31ea4c, 44
290 data81eaeaeaeaeaeaeaeaae00c1b
    d0bc18d20d08d21d0bd15c18dacc0
    e8e009d002, 247
300 dataa2008e00c1bd01c18d73c04cf
    fc0eaeaad01dca20020bdc0ad00dc
    a202a82901, 91
310 datad00fbd01d038e904c952b002a
    9524ce2c0982902d00fbd01d01869
    04c9c39002, 25
320 dataa9c39d01d0604c20c1eabdf7c
    049ff9df7c0fef7c060ea00000000
    eaeaeaea60, 206
330 data032e3640518cd9ecf6fe000f0
    c0b00000b0c0f0000ffffffffffff
    ffb0e6ff00, 152
340 dataa200ad06d0186df7c08d06d09
    001e8adf7c01001cae000f008ad10
    d049088d10, 38
350 datad0ad07d0186df8c0a8c952b00
    aa052a20120eac04c60c1c9c49005
    a0c44c4fc1, 97
360 data8c07d0eaad1ed0a82908f024a
    200ad10d02908f001e8adf7c03001
    cae000d010, 48
370 data20eac0a221982901f002a2388
    e06d0eaad10d02908f02cad06d0c9
    f0b025c9e0, 74
380 data9010ad10d029f78d10d0a900e
    ef9c04cbbc1c968900deefac0a958
    8d06d0a200, 145
390 data20eac0604a475644424f53434
    82055545245434854203033302039
    3436313630, 83
400 data0,60,0,0,102,0,0,195,0,1,
    153,128,3,255,192,15,255,240,
    24,0,24
410 data48,0,12,96,126,6,192,195,
    3,127,153,254

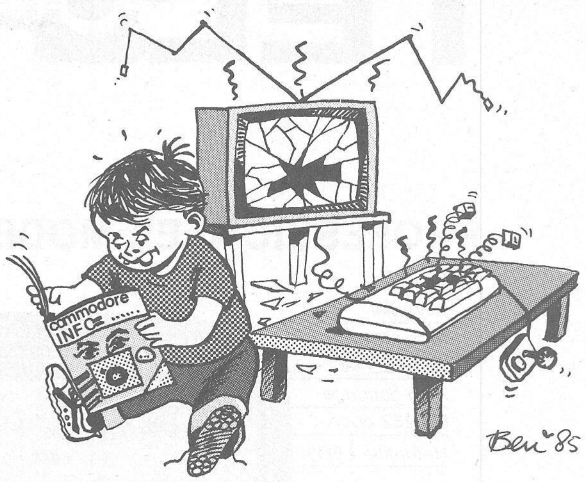

*** EINDE LISTING ***
```

```
crazy sprite


regel 1    135
regel 2    26
regel 3    215
regel 4    143
regel 20    151
regel 30    224
regel 40    171
regel 50    115
regel 60    177
regel 70    33
regel 80    84
regel 90    106
regel 100    205
regel 110    181
regel 120    101
regel 130    154
regel 140    180
regel 150    163
regel 160    105
regel 170    91
regel 180    61
regel 190    175
regel 200    145
regel 210    217
regel 220    243
regel 230    87
regel 240    209
regel 250    119
regel 260    42
regel 270    9
regel 280    62
regel 290    84
regel 300    148
regel 310    29
regel 320    3
regel 330    30
regel 340    54
regel 350    108
regel 360    73
regel 370    78
regel 380    198
regel 390    70
regel 400    15
regel 410    224

ready.
```

## C-16 DATABASE




We zijn iedere keer weer blij met goede C-16 / Plus-4 listings, waarmee het grote tekort aan goede software wordt verminderd. Dit programma, ook al een inzending voor de prijsvraag, geeft in heel bepekte ruimte een vrijwel compleet database programma, waarmee tot 100 records, (het woord bestand is in het programma niet juist gebruikt) kunnen worden ingegeven en op een disk weggezet. Alleen voor diskbezitters, maar wel erg fraai, compleet met duidelijk keuzemenu.

```
10 COLOR0,1:COLOR4,1:CLR:P$="                              ":GOTO30
20 GETKEYU$:RETURN
30 X=INT(FRE(0)/89):DIMF$(X+1):H=1
40 PRINT"[CLR-HOME][C-HOOG][WIT]"SPC(10)"N.A.W. BESTAND (C)"
50 PRINTSPC(10)"=================="
60 PRINT"[C-2*NEER] 1 =AANMAKEN BESTAND - MAX.";X:S=INT(X+1-H)
70 PRINT"[C-NEER] 2 =WYZIGEN BESTAND"
80 PRINT"[C-NEER] 3 =SORTEREN BESTAND":PRINT"[C-NEER] 4 =PRINTEN BESTAND"
90 PRINT"[C-NEER] 5 =BESTAND BEWAREN OP DISK"
95 PRINT"[C-NEER] 6 =BESTAND OPVRAGEN VAN DISK"
100 PRINT"[C-NEER] 7 =UITVOER GEGEVENS OP SCHERM"
110 PRINTSPC(4)"[2*NEER]NOG TE MAKEN BESTANDEN";S
120 PRINTSPC(4)"=========================="
130 PRINTSPC(4)"[C-NEER]AL GEMAAKTEN BESTANDEN";H-1
140 PRINTSPC(4)"=========================="
150 GOSUB20:IFVAL(U$)<1ORVAL(U$)>7THEN150
160 ONVAL(U$)GOTO240,170,260,300,470,500,790
170 PRINT"[CLR-HOME][C-NEER]VOER IN DE TE WIJZIGEN NAAM":PRINT"[C-2*NEER] MAX.19
 TEK."
180 GOSUB550:IFB=XORA$="*"THEN40
190 SAVE,1,10,"U DIENT GEGEVENS OPNIEUW IN TE VOEREN":FORI=1TO999:NEXTI
200 PRINT"[CLR-HOME] NAAM"SPC(10);MID$(F$(B),1,19):PRINT" ADRES"SPC(9);MID$(F$(B
),20,19)
210 PRINT" POSTCODE"SPC(6);MID$(F$(B),39,8):PRINT" PLAATS"SPC(8);MID$(F$(B),47,1
4)
220 PRINT" GEB.DATUM"SPC(5);MID$(F$(B),61,8):PRINT" TEL.NR."SPC(7);MID$(F$(B),69
,11)
230 GOSUB660:F$(B)=F$(H):GOTO40
240 IFH-1=XTHEN40
250 H=H-1:T$="":DOUNTILH-1=(X)ORT$="*":H=H+1:GOSUB640:LOOP:GOTO40
260 CHAR,22,8,"SORTERING":W=H-1
270 FORJ=1TOW-1:FORK=J+1TOW
280 IFF$(J)>F$(K)THENS$=F$(J):F$(J)=F$(K):F$(K)=S$
290 NEXTK:NEXTJ:GOTO790
300 PRINT"[CLR-HOME]                  PRINTEN":PRINT"[C-2*NEER] 1 =ETIKETTENLIJST"
310 PRINT"[C-NEER] 2 =VERJAARDAGLIJST":PRINT"[C-NEER] 3 =ALGEMENELIJST"
320 PRINT"[C-NEER] 4 =PERSOONS LIJST PRINTEN"
330 PRINT"[C-NEER] 5 =TERUG NAAR MENU"
340 GOSUB20:IFVAL(U$)<1ORVAL(U$)>5THEN340
350 ONVAL(U$)GOTO360,390,420,520,40
360 OPEN2,4:FORI=1TOH-1:PRINT#2,MID$(N$(I),1,19)
370 PRINT#2,MID$(F$(I),20,19)
380 PRINT#2,MID$(F$(I),39,8),MID$(F$(I),47,14):PRINT#2:NEXTI:CLOSE2:GOTO40
390 OPEN2,4:PRINT#2,"NAAM"SPC(21)"GEBOREN OP"
400 PRINT#2,"===="SPC(21)"=========="
410 FORI=1TOH-1:PRINT#2,MID$(F$(I),1,19);SPC(6);MID$(F$(I),61,8):NEXTI:CLOSE2:GO
TO40
```

```
420 OPEN2,4:FORG=1TOH-1:M=I:GOSUB430:PRINT#2:NEXTG:CLOSE2:GOTO40
430 PRINT#2,MID$(F$(M),1,19);SPC(6);MID$(F$(M),20,19)
440 PRINT#2,MID$(F$(M),39,8);SPC(17);MID$(F$(M),47,14)
450 PRINT#2,"GEB.DAT:";MID$(F$(M),61,8);
460 PRINT#2,SPC(9)"TEL.NR:";MID$(F$(M),69,11):RETURN
470 IFH-1<3THEN40
480 CHAR,29,12,"SAVING":OPEN3,8,2,"@0:ADRESSEN,S,W":PRINT#3,H
490 FORF=1TOH-1:PRINT#3,F$(F):NEXTF:CLOSE3:GOTO40
500 SAVE,31,14,"LOADING":OPEN3,8,2,"ADRESSEN,S,R":INPUT#3,H
510 FORE=1TOH-1:INPUT#3,F$(E):NEXTE:CLOSE3:GOTO40
520 PRINT"[CLR-HOME][C-NEER] VOER IN TE PRINTEN NAAM":PRINT"[C-2*NEER] MAX.19 TE
K."
530 GOSUB550:IFB=XTHEN40
540 GOTO630
550 SAVE,13,4,"":INPUTA$:V=LEN(A$):C$="":IFA$="*"THENRETURN
560 B=0:DOUNTILA$=C$ORB=(X)
570 B=B+1:C$=MID$(F$(B),1,V):CLOSE
580 IFB=(X)THENCHAR,1,10,"NIET GEVONDEN":FORD=1TO1200:NEXTD:RETURN
590 CHAR,8,6,"NAAM :":CHAR,15,6,MID$(F$(B),1,19)
600 CHAR,1,8,"IS DIT JUIST? (J/N)":GOSUB20
610 IFU$="J"THENRETURN
620 CHAR,1,8,F$:GOTO550
630 OPEN2,4:M=B:GOSUB430:CLOSE2:GOTO40
640 PRINT"[CLR-HOME] NAAM":PRINT" ADRES":PRINT" POSTCODE":PRINT" PLAATS"
650 PRINT" GEB.DATUM":PRINT" TEL.NR.":IFH-1=(X)THENRETURN
660 F$(H)="":RESTORE940:FORA=1TO6:READY
670 L$="":CHAR,12,A,"":INPUTT$:CHAR,0,20,F$
680 IFT$="*"THENRETURN
690 IFLEN(T$)>YTHENCHAR,1,20,"TE VEEL TEKENS":GOTO670
700 IFLEN(T$)=YTHEN730
720 FORC=1TOY-LEN(T$):L$=L$+CHR$(32):NEXTC
730 F$(H)=F$(H)+T$+L$:NEXTA
740 CHAR,0,15,"AKKOORD (1=JA/2=NEE)"
750 CHAR,0,17,"TYPE (*) VOOR TERUG IN MENU":GOSUB20
760 IFVAL(U$)<1ORVAL(U$)>2THEN740
770 ONVAL(U$)GOTO780,660
780 RETURN
790 B=0
800 B=B+1:PRINT"[CLR-HOME]   INGEVOERD BESTAND NR.";B
810 PRINT"[C-3*NEER]NAAM":PRINT"ADRES":PRINT"POSTCODE":PRINT"PLAATS"
820 PRINT"GEB.DATUM":PRINT"TEL.NR."
830 R=5:DOUNTILR=11:CHAR,15,R,":":R=R+1
840 LOOP:RESTORE940:G=1:Z=5
850 E=1:DOUNTILE=7:READY:G$=MID$(F$(B),G,Y):G=G+Y
860 CHAR,17,Z,G$:E=E+1:Z=Z+1
870 LOOP:CHAR,0,19,"TOETS 'SPACE  VOOR HET VOLGENDE ADRES"
880 CHAR,0,21,"TOETS 'M  VOOR MENU":GOSUB20
900 IFU$="-"THENB=B-2
910 IFU$="+"THENB=B+9
920 IFU$="M"THENB=X
930 IFB<H-1THEN800
935 GOTO40
940 DATA19,19,8,14,8,11
READY.
```

# Boekenrubriek

## 40 Grafische Programma's voor de C-64

**M.Sutter, *f* 29.50**

Dit boek, met als subtitel **Leer programmeren met hoge resolutie graphics in Basic**, geeft een serie listings, waarbij met hulp van de Hires-mode van de C-64 de nodige al of niet wiskundige figuren op je scherm worden getoverd. Hetzelfde boek is onlangs verschenen voor de MSX-computers, en dat blijkt ook uit de in het boek gebruikte Basic-syntax. Er wordt vanuit gegaan, dat de C-64 gebruiker beschikt over **Simons Basic**, zodat de commando's HIRES en LINE kunnen worden toegepast. Voor degenen die niet over deze utility beschikken is achter in het boek de Basic-versie van deze commando's opgenomen, maar dat betekent dat je bij het runnen wel een paar minuten geduld moet hebben, voordat er ook maar iets op je grafische scherm verschijnt.

De 40 listings zijn over het algemeen niet langer dan zo'n 50 regels Basic, dus vrij snel in te toetsen. Het eerste deel behandelt de meer wiskundige figuren, zoals drie-dimensionale driehoeken, moiree-effecten en ingeschreven vierkanten.

Verder krijg je sinuskrommen, parabolen en andere cartesische vormen voorgeschoteld. Histogrammen en geografie zijn zelfs niet vergeten, en bij iedere toepassing vind je één of meer illustraties van het grafische scherm, zodat je voordat je aan een programma begint alvast kunt bekijken hoe de figuren eruit gaan zien.

Als appendix zijn een paar (grote) machinetaal-programma's opgenomen, die zijn overgenomen uit het Amerikaanse tijdschrift Commodore Computing, waarmee de nodige Hires-opdrachten door machinetaal kunnen worden vervangen.

Voor degenen onder ons, die wat meer willen doen met de grafische mogelijkheden van de 64, een goed boek om mee van start te gaan. Wel is het aan te bevelen, om hierbij toch Simons Basic te gebruiken. Beschik je over een meer dan gemiddelde kennis van de 6510 machinetaal, dan zijn de assembly-listings achter in het boek duidelijk genoeg om zelf erg snel in Hires te kunnen gaan werken.

**Voor de prijs de moeite waard.**

## Werken met de Commodore 64

**Peter, SMD**

Een "originele" titel, bedacht door de heren P.Peters en H.Peters, de auteurs van dit boek. Uitgegeven door SMD, Leiden.

De eerste 100 pagina's van dit werk worden gewijd aan de beschikbare Basic-opdrachten en -commando's. Naast een korte uitleg, bedoeld voor de lezers op nulnivo, krijg je voor ieder commando een paar voorbeelden en opgaven. Deze zijn niet altijd even duidelijk en goed gekozen. Complete listings, die meerdere opdrachten verenigen tot een nuttig programma zul je hierin vergeefs zoeken.

Het tweede deel van dit boek geeft een nederlandse handleiding voor standaard software als de tekstverwerker EasyScript, C-64 BASE bestandsverwerking en het rekenmatrix/ calculatieprogramma ABRA-CALC. Heb je (toevallig) één van deze pakketten en ben je de originele gebruiksaanwijzing kwijt, dan kan dit boek uitkomst bieden.

Het boek zou goed kunnen dienen als leerboek voor een computer-cursus, waarbij de nodige persoonlijke begeleiding het geheel wat kan verduidelijken. Bezit je het C-64 handboek en dito boeken voor je software, of weet je al het één en ander van Basic, dan kun je dit boek rustig in de winkel laten liggen.

## Doe het zelf programma's op de COMMODORE-64

D.Kreutner *f* 39,90

Dit door Academic Service uitgegeven boek ligt sinds mei 1985 in de boekhandel. In tegenstelling tot de meeste andere boeken over dit onderwerp zitten er in dit boek nogal wat nuttige listings, waarmee je alle kanten uitkunt, ook zonder het naadje van de Basic-kous te kennen. Niet alles is even origineel, maar dat kan ook niet voor 40 gulden. De meeste programma's zijn voorzien van een duidelijke uitleg, waar het de kernroutines betreft, zodat het niet zo moeilijk wordt om de programma's aan te passen aan de eigen behoefte.

Het boek is bedoeld voor de C-64 bezitters, die werken met een cassette-recorder, maar de meeste listings zijn voorbereid om over te zetten naar gebruik van een diskdrive en (waar nodig) de printer. Verwacht geen sterke staaltjes van flitsende machinetaal, het is alles puur en rudimentair Basic wat de klok slaat. Wat te denken van het programma PIEP1 waarbij je steeds een boodschap op je scherm print, die vergezeld wordt van een aantal door de gebruiker in te geven piepjes, (PIEP2 is er ook natuurlijk), of KRIJGERTJE, waarbij je in slow-motion een arcadespel kunt spelen.

De **zakelijke** toepassingen hebben wat meer aandacht gekregen, en de serieuze beginners kunnen aan de hand van dit boek vrij goed het werken met sequentiële bestanden onder de knie krijgen. De relatieve bestanden worden hierbij (helaas) overgeslagen.

Het boek eindigt met enkele Basic-programma's, die samen de opzet vormen voor een (simpele) boekhouding.

Hoewel in Nederland in 1985 uitgegeven, vinden we de stand van zaken terug in dit boek, zoals die in 1983 bestond, rond de C-64. Er wordt als randapparatuur b.v. nog verwezen naar de printers die voor de VIC-20 bestemd waren en het ook wel deden op de 64 en zo nog wel wat zaken, maar de Basic-syntax is in de jaren niet veranderd, dus op zich hoeft dat geen beletsel te zijn, maar toch..... ●

## INZENDVOORWAARDEN

Inzendingen van programma's naar Commodore-Info zijn zeer welkom, maar denk er wel aan altijd een bandje of floppy in te sturen en in de listing geen grafische tekens of moeilijke commando-strings te gebruiken, die leiden te snel tot problemen bij het afdrukken. De listing op papier sturen hoeft niet, maar geef wel aan hoeveel regels het zijn en wat het ongeveer doet en welke tekst er bij zou moeten.

Adresseren aan PB 112, 1260 AC Blaricum met vermelding **COMMODORE-INFO listing.**

Bij plaatsing ontvangt u een vergoeding, over het afwikkelen ervan krijgt u bericht, maar bel anders even (02152-65695) om dat te regelen, als u uw listing in het blad ziet. We betalen ongeveer 40 tot 65 gulden per pagina geplaatste listing (afhankelijk van de kwaliteit), maar vinden te lange listings niet erg interessant. Per nummer reiken we meestal ook een stuk hardware (printer/ monitor/diskdrive) uit aan de beste listing, maar dan moet de kwaliteit ook wel echt prima zijn.

Inzending (en plaatsing) houdt in, dat u alle rechten (dus ook voor de Infolist-cassette) aan ons overdraagt en dat u ons vrijwaart voor aanspraken van derden, dus als het programma niet origineel is, hetgeen helaas wel eens voorkomt. Een en ander betekent ook, dat u het programma niet meer kunt opsturen naar andere bladen of radioprogramma's.

Een kleine batterijvoeding voor een module met RAM geheugen, zeker met de energiezuinige CMOS RAM chips, daarmee zouden dergelijke geheugenschips ook bruikbaar worden als permanent of semi-Rom geheugen. Een nieuw soort insteekmodule komt tot ontwikkeling.

# 8 KBYTE CBM-64 BATTERIJ-GEHEUGEN

# SOFT-PROM




**Kan een batterij-geheugen met stuursoftware het programmeren van Eproms gaan vervangen?**

**De ontwikkeling van de RAM opent steeds meer perspectieven. In de eerste plaats zijn de prijzen van RAM geheugen erg gezakt de laatste jaren. Veel computers hebben nu standaard al 64, 256 of meer KB aan RAM en daarmee zijn bv. RAM-disk toepassingen heel populair geworden. De ellende bij RAM is, dat wanneer de stroom wegvalt het geheugen weer leeg is. Maar via een klein batterijtje en gebruik van energiezuinige RAM's is ook dat wel te verhelpen.**

Van Zero uit Ridderkerk is er nu een produkt, dat voor de 64 die permanente RAM in de vorm van een module binnen bereik brengt.
Men noemt het de Soft Prom, omdat het qua funktie met ook een PROM, een **Programmeerbaar Leesgeheugen** te vergelijken is.
De Soft Prom is een 8 KBytes geheugen met extreem laag stroomverbruik, zo laag dat een 9 Volts batterijtje met een speciale schakeling, de inhoud 2 jaar bewaart, tenminste dat is de specificatie. Het geheel is in een insteekmodule ondergebracht (zie foto).

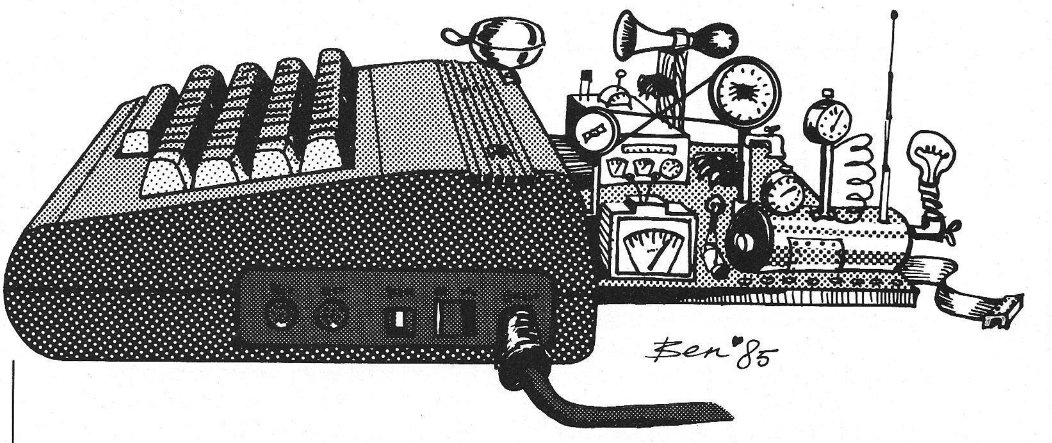
## De module

De SOFT-PROM wordt in het uitbreidingsslot van de CBM64 gestoken. Op de module bevinden zich twee schakelaars en een lampje. Met de ene schakelaar kan de module beveiligd worden tegen verder beschrijven van de inhoud en met de andere schakelaar kan de module van het CBM-64 geheugen worden losgekoppeld. Het lampje is de Aan/Uit indikatie. Het batterijvakje is niet erg gemakkelijk open te krijgen, maar dit is eigenlijk maar goed ook, anders zou het per ongeluk open kunnen gaan. Overigens is er een zgn. buffercondensator die voldoende stroom levert om

Het onderwerp van deze aflevering is de VIC Chip (6567 Video Interface Chip), die verantwoordelijk is voor het beeld op het scherm van de Commodore 64. VIC heeft veel mogelijkheden en laat zich ook doelmatiger in machinetaal dan in Basic programmeren. Tenzij je het extra Machinetaal-programma van deze keer gebruikt, want dat geeft je de beschikking over 5 nieuwe Basiccommando's, waaronder PLOT.

# Tekst en grafiek met de VIC-chip.

**Door Sjoerd Bakker.**



## Aflevering 7:

**Omdat het programmeren van VIC wat ingewikkelder gaat dan we bij de tot nu toe behandelde chips hebben gezien, zullen we hem in meerdere afleveringen aan bod laten komen. Per aflevering zullen we dan een bepaald aspekt belichten, zoals deze keer tekst en grafiek. We zullen VIC niet per register behandelen maar per onderwerp. De reden hiervoor is, dat VIC's registers ($D000-$D02E) nogal chaotisch zijn ingedeeld.**

### Oog

VIC zou je het beste kunnen vergelijken met een elektronisch oog. Je kunt hem vertellen naar welke geheugenplaatsen hij moet kijken, en hoe hij de informatie die hij daar aantreft moet interpreteren. Deze informatie wordt vervolgens automatisch doorgezonden naar je beeldscherm, waar je dan bijvoorbeeld een pagina tekst ziet.

Je zou het ook zo kunnen zeggen: via VIC kun je de 64.000(!) puntjes bekijken waarbij je beeldscherm ieder puntje ook nog één van 16 verschillende kleuren geeft.

VIC: BANKS

| $DD00 BITS 1/0 | BANK | GEBIED |
|---|---|---|
| 0 0 | 3 | $C000-$FFFF |
| 0 1 | 2 | $8000-$BFFF |
| 1 0 | 1 | $4000-$7FFF |
| 1 1 | 0 | $0000-$3FFF |

VIC kan 16 Kilobyte tegelijk "zien". De Commodore 64 heeft (zoals de naam al zegt) echter 64K RAM-geheugen tot zijn beschikking. We moeten VIC dus vertellen naar welk deel van deze geheugenplaatsen hij moet kijken. Hiertoe is het RAM-geheugen van CIA 2's register $DD00 bepalend naar welke bank VIC kijkt. Vormen ze samen de waarde 3 (bits = 11), dan kijkt VIC naar bank 0 die loopt van $0000-$3FFF. Bij de waarde 1(bits =01) bank 2 ($8000- $BFFF); en bij de waarde 0 (bits 00) bank 3 ($C000-$FFFF)

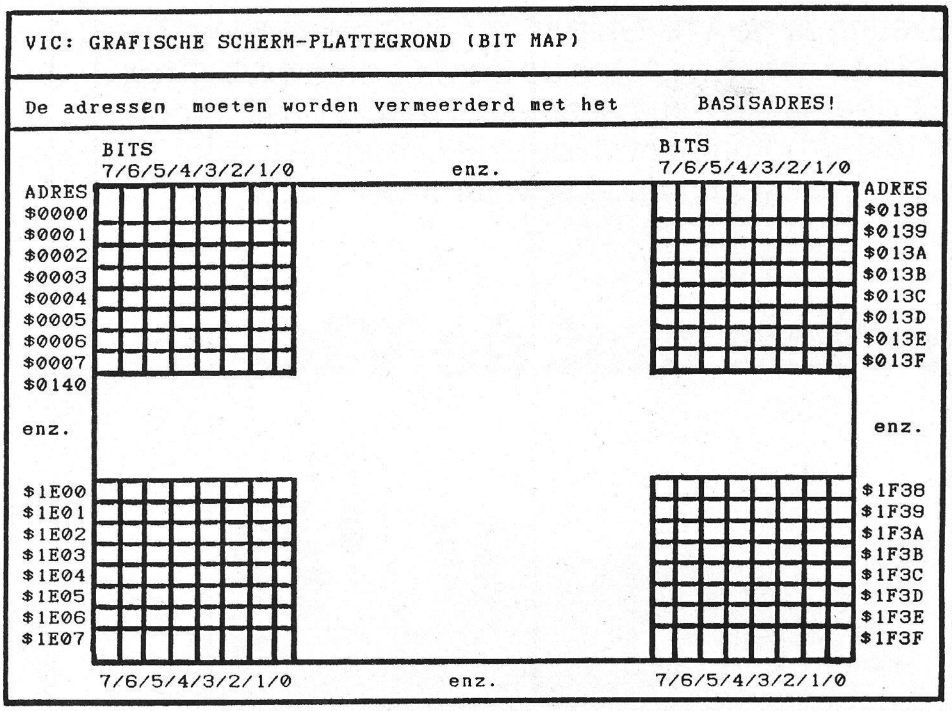
### Grafiek

VIC kan volgens twee verschillende hoofdprincipes werken. Hij kan het geheel van (64.000 bits/8=) 8000 bytes bekijken, en elk bit van dit geheel interpreteren als een punt op het beeldscherm dat "aan" (1), dan wel "uit" (0) moet worden gezet. In dat geval werkt hij volgens de "grafische mode". Hij kan echter ook een geheel van (8000/8=) 1000 bytes bekijken, en vervolgens iets anders doen met de informatie die hij daar aantreft. VIC staat dan in de "tekst-mode".

Dit is altijd het geval als je in Basic werkt zonder poke's te gebruiken. Maar ook spelletjes die veel grafische schermen bevatten, maken, paradoxaal genoeg, meestal gebruik van de tekst- mode.

Om te begrijpen hoe de twee modes werken, moeten we eerst weten hoe het beeldscherm is ingedeeld in de grafische mode. De grafische mode zet je aan door bit 5 van register $D011 "1" te maken. Omdat VIC 16K kan adresseren, en een grafisch scherm "maar" 8K beslaat, moeten we bepalen of we de onderste of bovenste 8K van de betreffende bank willen gebruiken. Dat gebeurt door bit 3 van register $D018 respektievelijk 0, dan wel 1 te maken. We hebben nu het basisadres van ons scherm. Als we bijvoorbeeld in bak 4 met de bovenste 8K willen werken, zal dit basisadres $2000 zijn.

Bijgaande illustratie laat zien hoe de geheugenplaatsen van het grafische gebied corresponderen met de puntjes waaruit het scherm is opgebouwd. Geheugenplaats $0000 (+ het basisadres!) wordt op het scherm voorgesteld door een horizontale rij van 8 puntjes, die uiterst linksboven in het scherm begint. De rij onder deze rij vormt een weerspiegeling van lokatie $0001. Zo gaat het door tot we 8 lokaties hebben gehad. Lokatie $0008 komt, weer horizontaal, rechts naast lokatie $0000 te liggen, etc. Als we helemaal rechts van het scherm zijn gekomen, zijn we toe aan lokatie (8x40=)320, oftewel $0140. Deze heeft betrekking op de 8 puntjes onder de




lokatie $0007. Dit patroon wordt het hele scherm door herhaald.
Simplistisch gesteld zou je kunnen zeggen dat elk 0-bit een "uit" voorstelt en elk 1-bit een andere, zodat er dus eigenlijk geen sprake is van aan en uit. In "multicolor-mode" krijg je weer een heel ander verhaal, maar daarover meer in de volgende aflevering.

## Tekst

Zoals gezegd staat de Commodore 64 als je in Basic bezig bent, meestal in tekstmode. De tekst mode zet je aan door bit 5 van register $D011 "0" te maken. Bij de bespreking van de grafische mode zal het misschien zijn opgevallen dat er telkens een vakje van 8 bij 8 puntjes op het scherm bij elkaar schijnt te horen. Dat klopt: een karakter of letterteken (we gebruiken die benaming door elkaar) beslaat namelijk de plaats van een matrix van 8 bij 8 puntjes. Zo'n vakje kun je in de grafische mode bit voor bit definiëren, maar in de tekstmode kun je met de informatie uit een enkele byte een heel vakje (8 bytes) vullen. Hiertoe maakt VIC gebruik van de z.g.n. Character-ROM. In deze ROM zijn de bitpatronen opgeslagen van alle tekens die de 64 standaard in huis heeft. Per teken bestaat zo'n bitpatroon uit 8 bytes, waarvan de bits zodanige waardes hebben dat ze op het scherm samen een herkenbaar karakter vormen.
Als bijvoorbeeld de inhoud van de lokatie $0400 (het normale basisadres van het schermgeheugen, oftewel screenmemory in tekst- mode) gelijk is aan $01, dan zal VIC het bitpatroon van de lokaties $01*8 t/m $02*8-1 (8-15) uit de Character-ROM op het scherm zetten. Dit zal als resultaat hebben dat er in het vakje linksboven een A verschijnt.
Elke byte van het schermgeheugen kan 256 verschillende waardes hebben ($00-$FF), die allemaal overeenkomen met een bepaald Character-ROM teken op het scherm. In de User Manual en Reference Guide staan lijsten met codes en de tekens die ermee overeenkomen.
De Character-ROM wordt door VIC op een speciale manier geadresseerd. In werkelijkheid loopt zijn adresbereik van $D000-$DFFF, waar ook de Commodore chips onderdak hebben. VIC ziet ze echter niet waar ze werkelijk zijn, maar ergens anders.

| CHARACTER-ROM HOOFDLETTER A | | |
|---|---|---|
| ADRES | BITS<br>7 6 5 4 3 2 1 0 | BYTE |
| $D008 | 0 0 0 1 1 0 0 0 | $18 |
| $D009 | 0 0 1 1 1 1 0 0 | $3C |
| $D00A | 0 1 1 0 0 1 1 0 | $66 |
| $D00B | 0 1 1 1 1 1 1 0 | $7E |
| $D00C | 0 1 1 0 0 1 1 0 | $66 |
| $D00D | 0 1 1 0 0 1 1 0 | $66 |
| $D00E | 0 1 1 0 0 1 1 0 | $66 |
| $D00F | 0 0 0 0 0 0 0 0 | $00 |

Hij kan ze alleen in de banken 0 en 2 zien en wel vanaf $1000 (set 1) of $1800 (set ") + het bank basisadres. Het komt er dus op neer dat hij ze in bank 0 kan zien vanaf $1000 of $1800 en in bank 2 vanaf $9000 of $9800.
VIC ziet ze op deze plaatsen, maar in werkelijkheid is hier slechts ongedefinieerd RAM-geheugen aanwezig dat, behalve voor andere VIC-funkties, verder normaal gebruikt kan worden. De originele bitpatronen vallen normaal gesproken natuurlijk buiten het adresbereik van de microprocessor, omdat in hetzelfde gebied de registers van de chips en de nybbles van het kleurengeheugen liggen.

## RAM-tekenset

VIC kan de karakterinformatie in plaats van uit de Character-ROM, ook uit RAM halen. Dit geeft de mogelijkheid om een zelfgemaakte karakterset te gebruiken. De totale Character- ROM beslaat 4K. We hebben telkens toegang tot maar een van de twee karakter-sets die erin zijn opgeslagen, wat neerkomt op 2K informatie per set. Er zijn dus (16/2=) 8 mogelijke 2K-gebieden waaruit VIC karakterinformatie zou kunnen halen. In bank 0 en 2 worden twee van die gebieden bezet door de ROM-bitpatronen, maar de overige kunnen gebruikt worden voor het benutten van een eigen karakterset. Bits 3-1 van register $D018 bepalen uit welk gebied VIC de karakterinformatie haalt. Bit 3 heeft dus een dubbele funktie : in de grafische mode bepaalt hij waar het scherm ligt en in de tekst-mode is hij het MSB (Most Significant Bit) voor aanduiding van de plaats van de karakterset.
Wellicht wordt het nu ook wat duidelijker hoe sommige spelletjes zo'n groot aantal **HIRES** (High Resolution)-schermen in de computer weten onder te brengen.
Ze maken een "oneigenlijk " gebruik van de mogelijkheid om met eigen karaktersets te werken, in de zin dat er geen lettersets, maar een of meerdere grafische sets worden ontworpen met symbolen, die goed passen bij het gewenste beeld.

**Met een eigen "grafische" karakterset kan men bij spelletjes meer doen met minder geheugen.**

Zo kun je achtergronden met bv. bakstenen ontwerpen en dan vrij eenvoudig en zonder veel geheugen te gebruiken als een soort tekstpagina's te vullen met letters, die geen letters zijn. In plaats van de normale 8K voor een grafisch scherm, afgezien van de kleur, heb je op deze manier maar 1K (25 regels van 40 breed) voor scherminformatie + 2K voor de karakterset = 3K voor een scherm nodig.

| VIC: PLAATS KARAKTER-SET | |
|---|---|
| $D018<br>BITS 3/2/1 | GEBIED<br>BANK BASISADRES + |
| 0 0 0 | $0000-$07FF |
| 0 0 1 | $0800-$0FFF |
| 0 1 0 | $1000-$17FF |
| 0 1 1 | $1800-$1FFF |
| 1 0 0 | $2000-$27FF |
| 1 0 1 | $2800-$2FFF |
| 1 1 0 | $3000-$37FF |
| 1 1 1 | $3800-$3FFF |

Gemiddeld kom je meestal nog lager uit, omdat met één set natuurlijk meerdere schermen samen te stellen zijn. Afgezien van de ruimtebesparing heeft bovenstaande methode nog een groot voordeel in verband met het z.g.n. "smooth scrollen" waarover we in de volgende aflevering meer gaan vertellen.

### Schermgeheugen

Bij werken in de tekstmode (hoe de tekenset er ook uit mag zien) haalt VIC de karaktercodes uit een gebied dat screenmemory oftewel schermgeheugen wordt genoemd. Zoals we al zagen vult hij met de informatie uit een schermgeheugenplaats "via" de karakterset een gebiedje van 8 bij 8 puntjes op het beeldscherm. Om het hele beeldscherm te vullen hebben we dus (8K/8 bytes) 1K schermgeheugen nodig. De bovenste vier bits (7-4) van register $D018 bepalen uit welke van de 16 mogelijke 1K gebieden VIC in tekst-mode de beeldinformatie haalt.
Lokatie $0000 (+ het basisadres) correspondeert met het vakje uiterst linksboven in het scherm. Lokatie $0001 is de rechterbuurman van dat vakje. Zo gaat het door tot we 40 geheugenplaatsen hebben gehad. Deze hebben dus betrekking op de bovenste schermregel. Lokatie $0028 (40 decimaal) is de onderbuurman van lokatie $0000, waarna er weer een regel met opklimmende waardes ontstaat.

| VIC: PLAATS SCHERMGEHEUGEN IN TEXT-MODE OF KLEURGEHEUGEN IN GRAFISCHE MODE | |
|---|---|
| $D018 BITS 7/6/5/4 | GEBIED BANK BASISADRES + |
| 0 0 0 0 | $0000-$03FF |
| 0 0 0 1 | $0400-$07FF |
| 0 0 1 0 | $0800-$0BFF |
| 0 0 1 1 | $0C00-$0FFF |
| 0 1 0 0 | $1000-$13FF |
| 0 1 0 1 | $1400-$17FF |
| 0 1 1 0 | $1800-$1BFF |
| 0 1 1 1 | $1C00-$1FFF |
| 1 0 0 0 | $2000-$23FF |
| 1 0 0 1 | $2400-$27FF |
| 1 0 1 0 | $2800-$2BFF |
| 1 0 1 1 | $2C00-$2FFF |
| 1 1 0 0 | $3000-$33FF |
| 1 1 0 1 | $3400-$37FF |
| 1 1 1 0 | $3800-$3BFF |
| 1 1 1 1 | $3C00-$3FFF |

Op die manier is verder het hele scherm samengesteld. Zie voor een hexadecimale plattegrond van het scherm- en het kleurgeheugen in tekst-mode Commodore Info no.4.

### Kleur

Om informatie op het scherm zichtbaar te maken heb je volgens de Wet van Bakker (nou, nou, red.) minstens twee verschillende kleuren nodig. In de meest simpele vorm werkt het dan zo, dat een 0-bit van door VIC gezien geheugen correspondeert met een bepaalde kleur die "background color" (achtergrondkleur) wordt genoemd, en een 1-bit met een andere kleur: de "foreground color" (voorgrondkleur).
Wat je dan op het scherm ziet is dus een afspiegeling van de toestand van 64.000 bits, wat ook wel een bit map (map=plattegrond) wordt genoemd.

De Commodore 64 heeft het vermogen om 16 verschillende kleuren zichtbaar te maken. Deze kleuren hebben elk een eigen code (0- 15). Omdat die keuze uit 16 codes met 4 bits (een nybble) gemaakt kan worden, kan een byte in principe worden gebruikt om de voor- en achtergrondkleur van een bepaald vakje weer te geven,

| VIC: KLEURCODES | |
|---|---|
| $00 | ZWART |
| $01 | WIT |
| $02 | ROOD |
| $03 | BLAUWGROEN |
| $04 | PAARS |
| $05 | GROEN |
| $06 | BLAUW |
| $07 | GEEL |
| $08 | ORANJE |
| $09 | BRUIN |
| $0A | LICHTROOD |
| $0B | GRIJS 1 |
| $0C | GRIJS 2 |
| $0D | LICHTGROEN |
| $0E | LICHTBLAUW |
| $0F | GRIJS 3 |

zpals we zullen zien bij het kleurgebruik in de grafische mode.
In de tekstmode maken we echter gebruik van de "Color-memory" om per vakje de voorgrondkleur aan te duiden, en van register $D021 om voor het hele scherm de achtergrondkleur te bepalen. Color- memory loopt van $D800-$DBFF, en deze geheugenplaatsen hebben betrekking op nybbles i.p.v. bytes. Dat wil zeggen dat bij adressering alleen de laagste 4 bits van de te verplaatsen waarde relevant zijn. De indeling is analoog aan het schermgeheugen, dus opklimmend in waarde van linksboven in het scherm naar rechtsonder. Color-memory kan worden verplaatst, en het adresgebied is dan ook voor elk scherm gelijk.

In de grafische mode wordt geen gebruik gemaakt van de Color- Memory. In plaats daarvan wordt de kleurinformatie per vakje van 8 bij 8 puntjes gehaald uit het gebied dat door register $D018 in tekst-mode als geheugen wordt toegewezen. De hoge nybbles in dit gebied bepalen voor elk vakje de voorgrondkleur, en de lage nybbles de achtergrondkleur. De indeling op het scherm is uiteraard hetzelfde als in de tekst-mode.
Hoewel je volgens de tot nu toe behandelde methodes al een aardig gevarieerd plaatje kunt maken, word je toch nog behoorlijk beperkt door het feit dat je per vakje maar twee kleuren kunt gebruiken , waar dan ook nog dat hele vakje aan vast zit. In de volgende aflevering zullen we ons dan ook o.a. bezig gaan houden met de "multicolor-mode", die je bij tekst, grafiek, en sprites kunt gebruiken.

## Border:randkleur

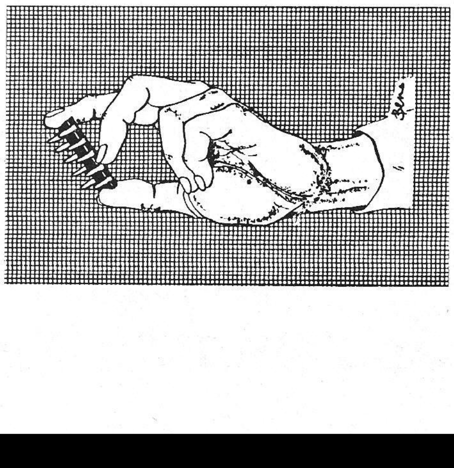
De inhoud van register $D020 bepaalt welke randkleur het scherm krijgt. Het is mogelijk om deze kleur aan het hele scherm te geven door bit 4 van register $D011 "1" te maken. Hierdoor wordt natuurlijk alle informatie op het scherm onzichtbaar. De processor wordt er echter wat door ontlast, zodat deze mogelijkheid bij alle toepassingen die een grote snelheid eisen, zoals spraak, vaak wordt gebruikt. Door bit 4 "0" te maken krijg je het scherm weer onveranderd terug.

## Plotten

Ik heb de 6502-machinetaal geleerd op mijn eerste computer: een Acorn Atom. Deze zeer eigenaardige machine is voorzien van een ingebouwde assembler die je kunt combineren met Basic- programma's. Verder zijn er een aantal grafische commando's aanwezig,zoals PLOT en DRAW. Daar tegenover staat dat je voor programma's slechts 512 bytes (!) RAM-geheugen tot je beschikking hebt.
Bij de Commodore 64 is het eigenlijk allemaal andersom. Geheugen is er meer dan genoeg, maar qua mogelijkheden voor beeld en geluid is het juist minder aantrekkelijk qua commando's en voorzieningen. Tenzij door POKE's natuurlijk, maar dat komt neer op het schrobben van de vloer met een tandenborstel. Daarom is het niet verwonderlijk dat programma's als **Simon's Basic**, die zulke commando's wel mogelijk maken, zich in zo'n grote populariteit mogen verheugen.

## Programma van deze maand

Maar ...wat grotere softwarehuizen kunnen, kunnen wij natuurlijk ook, en dat bewijzen we met het MT-programma van deze keer. Het maakt de 5 nieuwe Basic commando's **CLEAR, TEXT, HIRES,PLOT** en **UNPLOT** mogelijk. Het is evenwel wat groter dan gebruikelijk in deze serie. We zullen het dan ook in drie afleveringen behandelen waarbij alleen de meest interessante routines aan bod zullen komen. Ik neem tenminste aan, dat het bijvoorbeeld wel bekend is hoe je 8K met nullen vult.
We zullen de nieuwe opdrachten en hun betekenis even doornemen. **CLEAR** (niet te verwarren met CLR) maakt het grafische scherm dat we gaan gebruiken schoon en wist dus alle eerder geplotte zaken.
**TEXT** schakelt van grafische mode terug naar tekst-mode, waarbij het scherm nog alles bevat wat voor omschakeling aanwezig was.
**HIRES** brengt je in de grafische mode, waarbij ook weer alles bewaard is gebleven.



```
1 FOR A=40448 TO A+415
2 READ Y: POKE A,Y: NEXT
3 SYS 40488
10 DATA67,76,69,65,82,0,84,69
11 DATA88,84,0,72,73,82,69,83
12 DATA0,80,76,79,84,0,85,78
13 DATA80,76,79,84,0,144,158,192
14 DATA158,216,158,240,158,244,158,234
15 DATA234,120,169,64,141,8,3,169
16 DATA158,141,9,3,141,132,2,133
17 DATA52,133,56,88,96,234,234,234
18 DATA162,0,134,2,160,0,200,177
19 DATA122,201,32,240,249,16,3,76
20 DATA228,167,189,0,158,240,26,209
21 DATA122,208,4,200,232,208,243,232
22 DATA189,0,158,208,250,230,2,230
23 DATA2,232,224,29,208,214,76,228
24 DATA167,24,152,101,122,133,122,144
25 DATA2,230,123,166,2,189,29,158
26 DATA141,46,3,189,30,158,141,47
27 DATA3,108,46,3,234,234,234,234
28 DATA120,169,227,133,1,160,0,162
29 DATA220,132,251,134,252,169,230,145
30 DATA251,200,208,251,230,252,166,252
31 DATA224,224,208,243,169,0,145,251
32 DATA200,208,251,230,252,208,247,169
33 DATA231,133,1,88,76,152,159,234
34 DATA169,127,141,0,221,169,155,141
35 DATA17,208,169,20,141,24,208,76
36 DATA152,159,234,234,234,234,234,234
37 DATA169,124,141,0,221,169,187,141
38 DATA17,208,169,120,141,24,208,76
39 DATA152,159,234,234,234,234,234,234
40 DATA169,1,208,2,169,0,133,10
41 DATA32,235,183,224,200,176,14,165
42 DATA21,164,20,201,1,144,9,208
43 DATA4,192,64,144,3,76,72,178
44 DATA169,0,133,251,169,224,133,252
45 DATA169,64,133,253,169,1,133,254
46 DATA134,2,70,2,70,2,70,2
47 DATA56,169,24,229,2,168,32,128
48 DATA159,169,8,133,253,198,254,165
49 DATA20,72,70,21,102,20,70,20
50 DATA70,20,164,20,32,128,159,138
51 DATA73,7,41,7,133,253,160,1
52 DATA32,128,159,104,41,7,168,56
53 DATA169,0,106,136,16,252,200,120
54 DATA162,228,134,1,166,10,240,6
55 DATA17,251,145,251,208,6,73,255
56 DATA49,251,145,251,162,231,134,1
57 DATA88,76,152,159,234,234,234,234
58 DATA240,16,24,165,251,101,253,133
59 DATA251,165,252,101,254,133,252,136
60 DATA208,241,96,234,234,234,234,234
61 DATA32,121,0,76,174,167,234,234
```

**PLOT** moet voorzien zijn van twee parameters: een X- en een Y coordinaat. Er kunnen waardes van 0 t/m 319 in de horizontale en waardes van 0 t/m 199 in de vertikale richting worden gebruikt. Het formaat is dus :
PLOT X,Y
waarbij X en Y natuurlijk ook (andere) variabelen kunnen zijn.
Het effekt is, dat een puntje met breedtecoordinaat X en hoogtecoordinaat Y op het grafische scherm wordt aangezet. Voor UNPLOT geldt bovenstaand verhaal ook, alleen wordt het betreffende puntje nu uitgezet. Ik heb de Basic-loader van lage regelnummers voorzien, zodat er vanaf regel 100 plaats is voor een eigen programma dat van de nieuwe commando's gebruik maakt. Bijgaand demo-programma plot de overbekende Sinus-kromme.

## Aanpak Plot

Bij de realisering van het PLOT programma komen we drie verschillende hoofdproblemen tegen.
- Plaatsbepaling van het grafische scherm.

○ Samenwerking met de Basic interpreter.
○ Omzetting van de PLOT coordinaten naar een geheugenadres in het grafische scherm.

In deze aflevering zullen we het eerste probleem behandelen, als ook een routine uit de interpreter die veel te maken heeft met de oplossing van het tweede probleem.

Het eerste probleem stelt ons dus voor de vraag: waar zetten we het 8K grote Hiresscherm neer, dat we nodig hebben als we in de grafische mode willen werken. Omdat in de Commodore 64 "onder ROM" altijd RAM-geheugen (het geheugen waar VIC naar kijkt) aanwezig is, zou het verspilling van Basic programmaruimte zijn als we het grafische scherm in bank 0 ($0000-$3FFF) of bank 1 ($4000-$7FFF) zouden zetten. In bank 2 ($8000-$BFFF) zouden we het scherm onder de interpreter-ROM ($A000-$BFFF) kwijt kunnen, maar dan moeten we voor het kleurgeheugen 1K programma-RAM opofferen, waar nog eens 4K bij komt om de bitpatronen van de Character-ROM te omzeilen. Het wordt dus bank 3 ($C000-$FFFF), waarbij het scherm onder de Kernal-ROM ($E000-$FFFF) komt te liggen, en het kleurgeheugen onder de CIA's($DC00-$DFFF).

## CHRGET

Een wat vreemd aandoende afkorting op het eerste gezicht, deze CHRGET (Character get), de naam van een kleine subroutine uit de interpreter. De interpreter ligt, zoals bekend, in ROM en beslaat de geheugenplaatsen $A000-$BFFF. CHRGET wordt echter meteen na het inschakelen van de computer naar RAM gecopieerd omdat hij gebruik maakt van een pointer. Een pointer bestaat uit twee veranderbare geheugenplaatsen, waarvan de inhouden samen een wijzer vormen naar de te adresseren geheugenplaats. De twee belangrijkste funkties van CHRGET zijn het wegzeven van (voor de computer!) nutteloze spaties en het bepalen of data al dan niet numeriek van aard is. Alle door de computer te verwerken opdrachten passeren CHRGET, zodat de pointer steeds wijst naar het huidige commando-karakter. Dit geldt zowel voor "direkt mode" als voor "programma mode ".

DE ROUTINE CHRGET

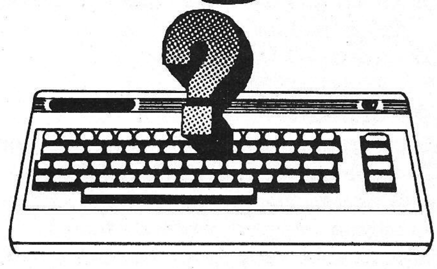
```
0073 E6 7A        INC $7A
0075 D0 02        BNE $0079
0077 E6 7B        INC $7B
0079 AD NN NN     LDA $NNNN
007C C9 3A        CMP #$3A
007E B0 0A        BCS $008A
0080 C9 20        CMP #$20
0082 F0 EF        BEQ $0073
0084 38           SEC
0085 E9 30        SBC #$30
0087 38           SEC
0088 E9 D0        SBC #$D0
008A 60           RTS
```

CHRGET begint op $0079. Daar wordt de inhoud van $9A, de LO-byte van de pointer, met 1 verhoogd. Mocht deze "op 0 zijn gesprongen", dan moet de Hi-byte $7B met 1 worden verhoogd. Zo niet, dan gaan we meteen naar $0097 waar trouwens de routine CHRGOT begint. Hier wordt de inhoud van het pointer-adres in de Accu geladen. Met CMP, BCS $008A wordt getest of de Accu-waarde groter of gelijk is aan $3A (de cijfers 0-9 hebben op ASCII-codes $30-$39). Zo ja, dan is de Carry 1, hebben we dus niet met een spatie of cijfer te doen en wordt de routine verlaten. Zo nee, dan wordt getest op $20 (spatie) en terug gesprongen naar het begin als dit het geval is en het volgende karakter in de opdracht wordt opgehaald.

Vervolgens wordt nog bekeken of de Accuwaarde kleiner dan $30 (dus ook geen cijfer"), maar op welke manier en waarom dit zo gebeurt mag de lezer zelf uitpuzzelen. We verklappen alvast dat CHRGET na aanroepen altijd wordt verlaten met een 0-Carry als Accuwaarde een cijfer voorstelt en met een 1-Carry als dit niet het geval is.

**Volgende keer gaan we verder met VIC, waar we flitsende sprites en ander ongeregeld goed zullen aantreffen. Ook zullen we weer een stap verder komen in de materie "Hoe maak ik eigen Basic-opdrachten en word ik toch niet overspannen?"** *S.B.* ●

# Vragen van Gebruikers.



## Nederlandse tekens

*P. uit de Bulten uit Amsterdam wil graag een methode om de nederlandse leestekens, dus bijvoorbeeld trema's, accent grave en accent aigu af te drukken vanuit z'n EasyScript. Hij bezit een daisywheel printer van het merk AVT80alfa. Hij vraagt zich af, of daar een mogelijkheid voor is, omdat op het letterwiel van z'n printer deze karakters wel aanwezig zijn.*

Helaas, om je teksten er puntgaaf uit te laten zien, kun je geen gebruik maken van C-64 tekstverwerkers als EasyScript en Vizawrite. Deze software is jammergenoeg niet zover aangepast dat ook afzonderlijke tekstdelen kunnen worden voorzien van speciale printeropdrachten. Wat wel mogelijk is, is het totale tekstbeeld in b.v. een &2&italic lettertype af te drukken. Om dat te doen moet je vóór het printen met hulp van F1 een bepaalde printerkode naar je printer sturen. Alle gegevens hierover vind je in het handboek van EasyScript. Op het ogenblik staat er een nieuwe versie van Superscript (Precision Software) op stapel, die meer mogelijkheden heeft als EasyScript en Vizawrite. We zullen hier ter zijner tijd ruim aandacht aan besteden.

## Schermbesturing cursor

*De heer A.A. van der Heijden uit Zoetermeer zoekt al maanden naar een mogelijkheid om de cursor en de schermbesturing van rechts naar links over het scherm te laten werken. Hij wil dit gebruiken in verband met een Hebreeuws letterset, dat hij heeft ontwikkeld.*

Het probleem van de cursorbesturing in de C-64 is nogal een complex geheel van machinetaal-routines, dat in de ROM zit ingebakken. Het is dan ook niet zo eenvoudig om in kort bestek uit te leggen, hoe dit in z'n werk moet gaan. Helaas hebben wij daar ook de tijd niet voor. Toch wil ik deze noodkreet hier plaatsen, in de hoop dat één van de lezers dit probleem aangrijpt om zelf op te lossen. Stuur je vondsten aan C-Info, en we zullen de beste oplossing zeker met publicatie honoreren. Wie helpt?

## Karaktergenerator

*Aad van de Kamer is enthousiast over het programma 'karaktergenerator' uit C-info nr 7. Hij wil graag weten, of het mogelijk is dit programma te veranderen, zodat het ook te gebruiken valt met een opslag op diskette in plaats van tape.*

Het betreffende programma heeft een korte saveroutine in machinetaal, waardoor de karakterset vanaf adres $3800 naar tape wordt geschreven als een (MT)programma. Dit programma krijgt als naam 'KARAKTERSET'. Om dit naar de drive te verplaatsen moet de data in regel 300 worden aangepast:

**300 DATA 169, 1, 162, 8, 160, 255, 169, 11, 162, 203, 160, 2, 32**

Daarna wordt dezelfde data naar een disk weggeschreven. Denk er wel om, dat er bij het saven steeds dezelfde naam wordt gebruikt, zodat het niet mogelijk is om meerdere sets op één disk te zetten. Wil je dit ook veranderen, dan moet je de data uit regel 320 na '96' aanpassen. Hier staat namelijk in ASCII 'karakterset'. Je moet er rekening mee houden dat het aantal bytes van de filenaam precies 11 moet zijn. Verder ben je vrij om voor elke set een andere naam te kiezen, als je de gegevens in regel 320 maar aanpast als ASCII-data. Om het karakterset weer in te laden type je eenvoudig LOAD "KARAKTERSET",8,1 en volgt verder de instrukties uit het programma.

## Knippertempo

*De cursor is momenteel erg aktueel. Dat blijkt uit de grote hoeveelheid vragen en tips die we binnenkrijgen over dit onderwerp. De aard van deze opmerkingen varieert van mensen die beweren dat ze migraine krijgen van dat geknipper, tot leuke truuks die je met de cursor kunt uithalen.*

Ik denk dat de hoofdpijn niet zozeer van de cursor komt alswel van een te langdurig kijken naar een onduidelijke monitor of slechte lichtomstandigheid. Om de cursor van het scherm te laten verdwijnen is in C-info 7 een oplossing gegeven. Een andere truuk is het vertragen of versnellen van dit knipperblokje. Dit kun je doen door op lokatie 56325 een bepaalde waarde te poken, waarop de cursor reageert. Deze waarde moet liggen tussen 0 en 255. Hoe lager dit getal, hoe sneller de cursor gaat knipperen en hoe sneller ook de repeteerfunkties kunnen worden gebruikt. Het tegendeel krijg je bij hoge getallen. Misschien is dit wat voor de migraine-lijders. Om de cursor weer te in z'n normale frekwentie te krijgen type je POKE 56325,62.

## Bytes free

*De vraag over het aantal Bytes die vrij zijn in het computergeheugen komt van Gerrit Hulst uit Assen. Hij kent het commando PRINT FRE(0), maar vindt het nogal eigenaardig, dat daar meestal een negatief getal komt te staan, wat volgens hem betekent dat er zelfs Bytes tekort zijn in het computergeheugen.*

Gerrit, als het erop aan komt heb je gelijk, alleen is de uitkomst van FRE(0) een grapje van Commodore. Ze hebben bij het ontwikkelen van de Basic-interpretor geen rekening gehouden met logisch denkende computeraars. Of, beter nog, Commodore is gewoon vergeten deze routine aan te passen. In de oudere PET-machines werkt de FRE-opdracht perfect, maar naarmate er meer (en ingewikkelder) RAM aan de computer werd toegevoegd, heeft Commodore de opdracht PRINT FRE(0) voor het gemak maar links laten liggen. De elementaire gegevens krijg je wel op het scherm, maar om er wijzer van te worden moet je zelf het één en ander aan deze opdracht toevoegen.

Om het aantal ongebruikte (vrije) Bytes van je VIC, C16, Plus/4 of C-64 geheugen te weten te komen moet je intypen:

PRINT FRE(0) - 65536*(FRE(0)<0)

Een andere manier is: PRINT FRE(0) + 2↑16

Om bij de C-64 te weten te komen hoeveel geheugen er door een (Basic)programma wordt bezet kun je typen: PRINT 38911-(FRE(0)-65536*(FRE(0)<0)

Hoewel enigszins gecompliceerd, geeft dit je uiteindelijk toch de juiste informatie over de staat waarin de computerRAM zich bevindt. Uiteraard kun je deze formules ook in je programma's gebruiken, om te bepalen hoeveel geheugen je nog beschikbaar hebt om gegevens in te kunne opslaan.

## Diskettegebruik



*Jean-Pierre van Vaerenbergh uit Affligem (Belgie) twijfelt tussen zuinigheid en betrouwbaarheid ten opzichte van z'n diskettes. Hij heeft ontdekt dat je een 5 1/4" disk best aan twee kanten kunt gebruiken, als je tegenover de bestaande inkeping nog een soortgelijke hoek uit de disk knipt. Daarmee kan hij de helft van z'n franken uitsparen, doordat er ineens twee maal zoveel gegevens op de disk kunnen. Toch blijft hij zitten met de vraag, of deze besparing geen keerzijde heeft.*

De kwaliteit van de diskettes die worden verkocht voor het gebruik met de Commodore 1541 (5 1/4 inch SSDD) is ten opzichte van een poos geleden nogal verbeterd. Daarbij is de prijs veel gunstiger geworden. Je kunt nu voor zo'n 4 a' 5 gulden al een redelijke diskette bemachtigen. Het enige waar je bij aanschaf op moet letten is de verstevinging van het centrale gat. Deze moet voorzien zijn van een "hub-ring", waardoor er aanzienlijk minder slijtage kan optreden. Om beschadiging van de disk te voorkomen, is aan de binnenzijde van de hoes een speciaal stof-filter aangebracht, waardoor het vuil als het ware wordt gevangen.
Zoals je -terecht- opmerkt, is de disk aan twee kanten voorzien van een magnetische laag, waarvan door de 1541 alleen de onderkant wordt benut. In principe is de andere kant ook te gebruiken, maar dat heeft tot gevolg dat de draairichting van de diskette wordt omgedraaid. Stof, vuil en andere ongerechtigheden worden op die manier tegen de vleug in weer uit het filter losgetrokken, en kunnen daardoor alsnog de disk beschadigen. Gezien de prijs van de diskettes en het risico wat je ermee loopt deze schijven tweezijdig te gebruiken adviseer ik om dit maar zo weinig mogelijk te gaan doen.

## Commodore Monitoren

*A.van Heteren uit Groningen vraagt welk verschil er bestaat tussen de monitoren van Commodore type 1701 en 1702.*

Tussen de types 1701 en 1702 bestaat geen enkel zichtbaar verschil. Commodore is in 1984 begonnen met de 1702-serie, omdat (denk ik) de kleurenbuis in de monitor door een andere fabrikant werd geleverd. Beide toestellen komen uit Japan, en ook bij vergelijking van de twee is er geen wezenlijk verschil. Wel zijn de 1701 types ouder dan de 1702, dus je hebt bij de eerste wat meer kans op een minder scherp beeld.
Naast deze monitoren maakt Commodore nog de 141 Color Monitor, die bedoeld is voor de Plus/4 & C16. Deze is, afgezien van de kleur van de kast, gelijk aan de 1702. Verder staan ons een paar nieuwe types te wachten, de monochrome 1901 en de 1902, die - naast de video/composiet, over een RGB(I) ingang beschikt. Deze monitoren zijn bestemd voor de 128, waarbij zowel een 40- als 80- koloms output op het scherm kan worden gebracht.

## String too long error

*Robin van Kralingen heeft een programma waarin data naar tape wordt geschreven, die als hij weer wordt ingelzen een STRING TOO LONG ERROR veroorzaakt. Hij kan de reden van deze fout niet vinden, maar kan tevens de gegevens niet terughalen uit z'n bestand.*

De fout die Robin kennelijk heeft gemaakt -en hij is zeker niet de enige- bij het wegschrijven van de gegevens naar tape (of disk) is het vergeten van een 'delimiter' ofwel een teken, dat de verschillende strings die worden weggeschreven van elkaar scheidt.
Als de gegevens dan weer worden teruggelezen, weet de computer niet waar de ene string ophoudt en de volgende begint. Hij probeert alle bytes in één string te stoppen, en dat gaat goed, totdat er meer dan 256 tekens in zitten. Dat heeft de STRING TOO LONG ERROR tot gevolg.
Maar hoe doe je het wél goed? De beste manier is, om achter iedere string die je wegschrijft naar cassette of disk een CHR$(13) mee te geven. Dit return-karakter fungeert dan als delimitor. Je kunt ook een komma gebruiken in plaats van de return, maar de eerste is algemener. Een voorbeeld om een bestand weg te schrijven en weer in te lezen vind je hierbij.

```
10 A$ = "Dit is eerste string"
20 B$ = "En dit de tweede string"
30 C$ = "De laatste string"
40 OPEN 2,1,1,"testfile" (disk : OPEN
   2,8,2,"test,s,w")
50 PRINT#2,A$;CHR$(13);B$;
   CHR$(13);C$
60 CLOSE 2
70 PRINT"Cassette terugspoelen. (typ
   toets)"
80 GET X$: IF X$="" THEN 80
90 OPEN 2,1,0,"tesfile" (disk : OPEN
   2,8,2,"test,s,r")
100 INPUT#2,D$,E$,F$
110 PRINT D$,E$,F$
120 CLOSE 2
```

Dit programma schrijft de strings A$,B$ en C$ op de juist manier naar tape (of disk) en leest ze vervolgens weer terug in andere variabelen. Hierdoor wordt de manier van lezen en schrijven hopelijk wat duidelijker.

## Verborgen C-64 RAM

*Sandra Mooiweer uit Den Haag is al een poos bezig met machinetaal. Ze wil graag weten of het mogelijk is om de 'schaduw-RAM' van de C-64 te gebruiken voor het opslaan van haar eigen assembler-programma's. Daarmee bedoelt ze de RAM onder de Basic-ROM. Als dat kan, wil ze ook graag informatie over het uitschakelen van de ROM, zodat ze met de RAM kan gaan werken.*

Beste Sandra, er zitten inderdaad nog precies 16384 (16K) Bytes verstopte RAM in de C-64. Je bent niet de eerste die dat ontdekt, er zijn zelfs al printjes te koop, die dit RAM-gebied beschikbaar maken voor het gebruik met Basic. Maar ik ben het met je eens, als je zegt dat het veel leuker is om deze foefjes zelf te ontdekken. 8KBytes van deze RAM zit onder de Basic-ROM op de adressen 40960 tot 49151 ($A000 - $BFFF). De andere Bytes vind je onder het Kernel-gebied op $E000 tot $FFFF ofwel in decimalen 57344 tot 65535.
Om deze RAM-gebieden te kunnen aanspreken moet je gebruik maken van een geheugenlokatie in de nulpagina, adres $01. Daarvoor moet de Basic en Kernel ROM worden uitgeschakeld en je eigen RAM worden aangezet. Op adres $01 staat een registervlag, die o.a. aangeeft welke gebieden er op een bepaald moment kunnen worden gebruikt. Normaal is de waarde van dit adres 55. Als je bit 0 op 0 zet, schakel je de Basic-ROM uit en RAM aan. Datzelfde geldt voor bit 1 die het Kernel-gebied bedient. In totaal beschik je dan over 16K RAM.
Om zowel Basic als Kernel uit te zetten type je: POKE 1, PEEK(1) AND 253
Wil je alleen het Basic gebied gebruiken, dan verander je de 253 in 254. Terugzetten gebeurt met POKE 1,55.
Deze RAM/ROMgebieden hebben wat eigenaardigheden, waar het PEEK en POKE betreft. Als je een PEEK geeft naar één van deze adressen, dan krijg je de waarden van Basic of Kernel, en niet van de onderliggende RAM. POKE je echter naar zo'n

adres, dan wordt automatisch de RAM veranderd. Op deze manier kun je gemakkelijk machinetaal kwijt in dit gebied, omdat je zowel in Basic als in machinetaal zondermeer via POKE deze plaatsen kunt bereiken. Wil je van hieruit werken dan moet je ook adres 1 op de juiste manier aanspreken. Sterkte, en we hopen snel een leuke toepassing van je te kunnen plaatsen.

## Foute software

*Het komt nogal eens voor, dat we brieven krijgen van gedesillusioneerde computeraars, die van hun laatste spaarcenten een nieuw spel hebben aangschaft, dat niet of niet goed blijkt te werken. Ee'n van die mensen is Dhr. Veldpaus uit Venray. Hij vraagt wat daaraan te doen is.*

Het is een trieste zaak, dat de softwareproducenten zo weinig kritisch omspringen met hun handelswaar. Helaas is meer dan 10 procent van de verkochte tapes ondeugdelijk. Dit komt voor een deel door de manier van kopiëren die de heren erop na houden. Daarbij is het natuurlijk ook voor hen ondoenlijk om iedere tape afzonderlijk te kontroleren. Wel zouden er wat meer steekproeven genomen moeten worden, waardoor het verval aanzienlijk zou verminderen. De fabrikant denkt er meestal wat te gemakkelijk over.
Aan de andere kant heeft ook de consument in het algemeen hier door z'n gedrag aan bijgedragen, want momenteel moet zoveel worden gedaan om de tape van piraterij te kunnen vrijwaren, dat de deugdelijkheid hieronder moet lijden. Daar hebben we hetzelf naar gemaakt.
De enige remedie hiertegen is teruggaan naar de verkoper en óf je geld terug eisen óf een nieuwe tape. Zover ik weet is iedere dealer en producent op de hoogte van de grote hoeveelheid uitval, en is dan ook zondermeer genegen ondeugdelijke tapes te vervangen. Adressen van importeurs en producenten vind je doorgaans op de verpakking.

## Printshop

*Het gebruik van een van de meest creatieve programma's voor de C- 64 'PRINTSHOP' blijkt soms nog wel wat problemen te geven in verband met het aansturen van de printer. En zonder printer kun je erg weinig met deze software. Ook A.de Leeuw uit Amsterdam kampt met deze problemen.*

Om iedereen die overweegt 'Printshop' aan te schaffen de nodige gegevens te verstrekken volgt hier een complete lijst van de printers en interfaces, waarmee dit programma kan worden gebruikt. De printshop-diskette is tweezijdig beschreven (!) waarbij de versie op kant 2 uitsluitend geschikt is voor de Commodore grafische printers MPS-801 en VIC 1525.
Kant 1 van de printshop-disk kan worden gebruikt met de volgende printers: Epson RX-80/MX-80 en Epson RX-100/MX-100, Epson FX- 80/100, Star Micronics 10X/ 15X, C-Itho 8510, NEC 8023A, Legend 880, Panasonic KX-P1090/1091, Mannesmann Tally Spirit 80, Blue Chip, Okidata 92/93. Als interface tussen C-64 en één van bovenstaande printers kun je gebruiken : Xetec, Cardco, Grappler CD Tymac connection, TurboPrint/GT of Micro World-350.
Bezit je een andere printer en/of interface, dan werkt dit programma waarschijnlijk niet. Maar je kunt natuurlijk altijd de proef wagen, door met je hele configuratie naar de winkel te stappen en het uit te proberen. Denk erom, dat van de hier genoemde interfaces de grafische (duurdere) versie moet worden gebruikt. Wat de printers betreft, Okidata geeft bij gebruik van de kleinere lettertypes erg slechte resultaten.
*Jan Bodzinga*

dracht tussen computer en drive niet optimaal. Dit is echter uitsluitend te wijten aan de langzame 1541 drive.
Van Commodore hebben we nu de beschikking gekregen over de zojuist gelanceerde **1570**, de enkelzijdige voorloper van de 1571, en daarmee komen we tot een heel andere resultaten.
Deze snelle drive, die fl.899.– kost, heeft een opslagkapaciteit van 170 Kbyte (CBM-formaat) is enkelzijdig en kan zowel met de C-128 als met de oudere Commodore computers worden gebruikt, die over een seriële uitgang beschikken.
Daarnaast kan deze drive de meest voorkomende CP/M formaten lezen en schrijven. Hierdoor wordt het mogelijk om bestaande CP/M software en data op het originele formaat in de C-128 te lezen en eventueel te verwerken.
Het meest opvallende aan deze drive is echter de **snellere verwerking**. Om een paar cijfers te noemen, het formatteren van een diskette in CBM-formaat duurt 20 seconden. Backups 1 minuut. Het laden van een Basic-programma van 120 blokken gebeurt in 12 sec. en het opstarten in CP/M duurt nog maar 25 seconden. De tijd die de 1541 voor deze bewerkingen nodig heeft zijn meer dan het drievoudige van deze uitkomsten. Een hele verbetering voor het C-128 systeem. Deze drive is precies wat de C-128 tot nog toe miste om een volwassen systeem te kunnen worden genoemd. De tijd nodig voor het laden en saven, schrijven en lezen vormt bij de verwerkingen geen belemmering meer voor een serieus gebruik.
De 1570 bepaalt zelf de hoogste snelheid waarmee de seriële overdracht kan plaatsvinden. Deze ligt natuurlijk lager voor een langzamer device dan voor de snelle 2MHz van de C128/80. Maar zelfs voor gebruik met de C64 of C-16 is aanschaf van deze drive te overwegen. Verder heeft de 1570 nog een paar aardige truuks in z'n software waardoor het geheel nog wat gebruiksvriendelijker wordt. Zoals gezegd, de meeste CP/M formaten kunnen worden gelezen. Wel verdient het de voorkeur om de software uiteindelijk op CBM-CP/M formaat te zetten, omdat daarmee de hoogste overdracht-snelheden kunnen worden bereikt. Daarnaast kan voor de 128-Basic gebruik worden gemaakt van een '**BOOT-sector**'. Dit betekent dat bij het opstarten van de C-128 automatisch via de BOOT het juiste programma van de diskette wordt ingelezen en gerund, als tenminste met hulp van de meegeleverde utility op de BOOT is aangegeven welk programma dit is. Zo niet, dan meldt de 128 zich met het gebruikelijke opstartscherm.
Als laatste nog de mogelijkheid van de 1570 om data te 'bursten' ofwel in gewoon nederlands 'te spuiten'. Deze optie kan alleen in machinetaal worden gebruikt, maar wat er dan gebeurt, is wonderbaarlijk. Met **BURST** kan namelijk zeer snel data worden overgeseind naar computer of drive. Dit is vooral handig bij het inlezen van data voor b.v. Hires schermen en derg. Het lezen van zo'n 64000 bits gaat binnen een paar tellen. Wel moet de seriële bus van beide apparaten van het 'snelle' type zijn. Kortom een fantastisch apparaat, die 1570. Daar hebben ook de C-64 gebruikers al een poos op zitten wachten.
Hoewel Commodore bij de C-128 in eerste instantie de dubbelzijdige 1571 heeft aangekondigd, wordt in Europa eerst de enkelzijdige 1570 op de markt gebracht. De 1570 heeft exact dezelfde mogelijkheden als de 1571, maar kan slechts de helft van de opslagkapaciteit van de 1571 bevatten. Daarnaast is het voor de 1570 natuurlijk ook niet mogelijk om de dubbelzijdige CP/M 3.0 formaten te lezen. Voor het overige zijn beide apparaten intern identiek. De 1570 lijkt uiterlijk bijzonder veel op de 1541, terwijl de 1571 een totaal nieuwe behuizing krijgt. Levering van de 1571 wordt (naar men hoopt) gestart medio 1986. We zullen in de komende maanden nog uitgebreid op deze nieuwe drive terugkomen.

## C-64 compatibel ?

Van veel (verontruste) C-64 gebruikers bereikten ons vragen over de bestaande C-64 software, interfaces en toolkits. Unaniem wilde men graag weten of deze te gebruiken waren op de C-128.
In het algemeen kun je ervan uitgaan dat letterlijk alle software en zeker alle Commodore-randapparaten werken met de C-128. Ook met de meeste toolkits die worden aangesloten op de expansiepoort (en de userpoort) hoef je geen problemen te verwachten. De enige problemen die je kunt krijgen zitten in de seriële printer interfaces en in een deel van de op disk beschermde software.
Wat de Parallel Centronics interfaces betreft, er zijn nogal wat inferieure knutselwerkjes op de markt, die empirisch zijn ontwikkeld aan de hand van de C-64 data-overdracht. Deze zullen niet werken op de C-128, zeker niet als je de 1570/71 wilt gebruiken met 'burst-mode'. De betere, en dus meestal duurdere interfaces hebben rekening gehouden met het Commodore-protocol, en werken zondermeer op de C-128. In het ergste geval moet je dus een ander interface aanschaffen.
Wat de software betreft, de meeste -ook beschermde- software werkt op de 1570 drive. Alleen die software, die gebruikt maakt van het interne diskdrive operating systeem van de 1541 loopt de kans op mislukking bij de 1570.

## CP/M 3.0

Met de toepassing van de 1570/71 drive gaat een wereld aan nieuwe mogelijkheden open voor de traditionele Commodore gebruiker. De mogelijk wordt geschapen om via de C-128 kennis te maken met een operating system dat van diskette wordt ingelezen en ook software die niet specifiek voor Commodore is geschreven, ligt nu binnen het bereik. Er zitten wel een paar addertjes onder het gras. De meegeleverde CP/M disk bevat lang niet alle CP/M utilities, die horen bij versie 3.0. Alleen de meest elementaire zaken zijn opgenomen. Dit is door Commodore opzettelijk gedaan, om de prijs van het hele C-128 systeem laag te houden. Voor ieder verkocht exemplaar moet namelijk een (fors) bedrag worden betaald aan de eigenaar van CP/M, Digital Research. Bij het volle 3.0 pakket zou deze prijs doorberekend worden aan de consument. Een verstandig besluit, omdat lang niet iedere C-128 gebruiker CP/M tot het uiterste zal willen benutten. Wil men de beschikking over een compleet CP/M 3.0 systeem, dan kan binnenkort de aanvulling compleet met de nodige CP/M literatuur,bij Commodore worden besteld.
De software voor dit systeem is alleen te koop op niet-Commodore diskformaat. Andere formaten kunnen worden gelezen door de 1570, maar niet door de 1541. Commodore gaat geen CP/M software leveren, maar is in onderhandeling met de grote softwarehuizen over een mogelijk-

*vervolg op pag. 87*

Het aantal inzendingen voor onze Computer Anno 1990 prijsvraag en de kwaliteit ervan overtrof onze stoutste verwachtingen. Nederland blijkt behoorlijk creatief bezig te zijn met computers.

# IDEE 1990

# Uitslag prijsvraag

**Bij het beoordelen van de inzendingen voor de prijsvraag over de computer anno 1990 vonden we maar één ding jammer: er waren zo veel schitterende inzendingen, waaraan veel zorg en aandacht was besteed, dat we ook hele mooie hebben moeten laten liggen. Onze ruimte is helaas maar beperkt. Prachtige tekeningen, collages, hele boekwerken werden ons toegestuurd, door inzenders van 6 tot 75 jaar, die vrijwel allemaal meer dan de moeite waard waren. We hebben in ieder geval met plezier de vele vondsten doorgenomen, die soms humoristisch, soms zeer serieus waren. Een paar gebruikten we als illustratie hierbij, maar dat is slechts een klein deel van de meer dan 100 inzendingen.**



En zelfs wanneer we ze allemaal zouden afdrukken, wat deden we dan bv. met het hoorspel van D. van Rossen uit Zaandam of de prachtige inzendingen van een Commodore-Info tijdschrift anno 1990. Maar ja, dat is natuurlijk het probleem bij iedere wedstrijd, waar de vorm van de inzendingen zo vrijgelaten is als bij onze ideeënwedstrijd.

## Beurzen

We hebben gelukkig wel een oplossing gevonden om tenminste een deel van de inzendingen aan een breder publiek te laten zien. Op de Commodore-Info beurs en de PCM Show hebben we ze op grote platen hardboard bevestigd en in de buurt van onze stand opgehangen. Veel mensen hebben ze bekeken en we kregen nogal wat commentaar. Men vond het over het algemeen een leuk initiatief, maar het bleek moeilijk over te brengen, dat we zelf een primitieve tekening van een 8 jarige misschien meer waardeerden dan een perfect afgewerkte uiting van een volwassene. Ook bij de prijstoekenning hebben we geprobeerd wel degelijk rekening te houden met de leeftijd van de inzenders.

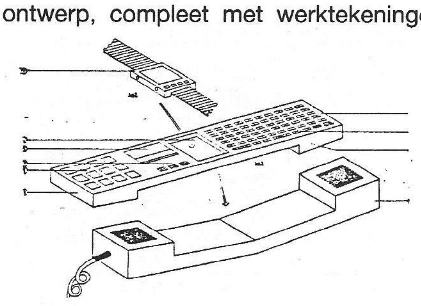
## Inzendingen

Bij de inzendigen waren erg veel mooie tekeningen, schema's en complete tijdschriften. Die laatste soms erg mooi uitgevoerd, men heeft door plakken, knippen en tekenen echt heel leuke computerbladen weten samen te stellen. We krijgen concurrentie in 1990! Het leuke is, dat blijkbaar meerdere inzenders op hetzelfde idee komen en zo waren er nogal wat ontwerpen voor de Commodore 256, een duidelijk vervolg op de 64 en 128. Misschien ook omdat ze graag de 128 wilden winnen? Ook robots waren een favoriet onderwerp en heel wat inzenders bedachten ook nieuwe toepassingen voor de polscomputer, daarin natuurlijk gestimuleerd door de uitgeloofde prijs, nl. een Seiko polscomputer.

## Uitslag

Het was werkelijk heel moeilijk en we zouden wel een lijst van potentiële winnaars van zo'n stuk of twintig kunnen maken. Maar ja, er moet toch één winnaar zijn. In overleg met de mensen van Commodore en Seiko, die tenslotte de prijzen uitloofden, is de keus gevallen op:

**Marc Hoos uit Den Bosch**
die een 128 krijgt voor zijn COMP2000 ontwerp, compleet met werktekeningen



M. Hoos

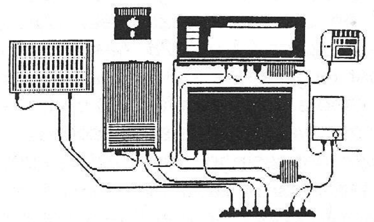
en uitgebreide toelichting, waarbij hij alle mogelijke computerfunkties combineerde en daarbij erg technisch en erg creatief te werk ging.

**Otto Tieleman uit Homberg**
Hij maakte een strip, 1990 of het leven van een spion, die niet alleen erg fraai is, maar ook een aardig verhaal bevat.




O. Tieleman

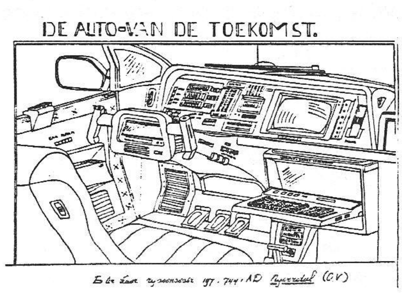
Geen grote prijs, maar wel een eervolle vermelding en een mooi softwarepakket als troostprijs, krijgen:



R. Kusters uit Arnhem



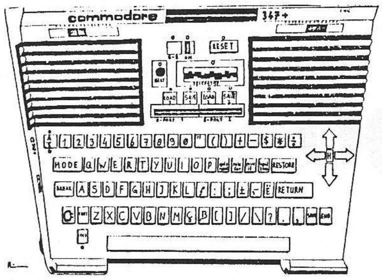

R. vd Woude uit Assen




B. ter Haar uit Nijverdal

R. Kusters uit Arnhem
R. vd Woude uit Assen
B. ter Haar uit Nijverdal
R. de Ruijter uit Rotterdam
D. van Rossen uit Schiedam

De prijswinnars krijgen bericht.




M. de Bruin

## Prijswinnaars

Toch blijven we zitten met een hele berg materiaal, waar we best iets meer mee zouden willen doen. Maar dat geldt ook voor de enorme hoeveelheid listings, die we uit de programma prijsvraag hebben overgehouden.
We hebben A en B bij elkaar opgeteld en besloten dit najaar nog een boek uit te brengen, waarin we de beste listings gaan afdrukken. En dan combineren we dat met de beste stukken uit de idee-prijsvraag. Uit alle inzenders, die hun bijdrage in de vorm van een tekening of in ieder geval een reproduceerbare vorm hebben ingezonden, hebben we een keuze gemaakt. We gaan die inzendingen gebruiken als illustraties in:

# Het grote COMMODORE-INFO Listingboek

Dit wordt een boek van zo'n 160 pagina's, waarin we ruim 80 programma's gaan plaatsen. De computers en computertoepassingen uit het jaar 1990 passen daar natuurlijk leuk bij. De verkoopprijs van het boek wordt *f* 24,90. De lezers van Commodore-Info kunnen echter bij voorintekening wat goedkoper dit boek bestellen, namelijk voor *f* 19,90 inclusief verzenden, zoals gebruikelijk door betaling op het gironummer 3157656 van Infolist. Een leuk idee voor de winteravonden, en het is natuurlijk ook als geschenk voor de feestdagen ideaal. We hebben het eind november klaar, dus heeft u het voor Sinterklaas in huis, wanneer u tenminste tijdig betaald hebt. In dit blad vindt u een bestelbon.
Iedere inzender, wiens werk we gebruiken, krijgt als beloning van ons een drietal exemplaren van het grote Commodore-Info Listingboek toegestuurd, zodat ze ook iets weg kunnen geven van wat waarschijnlijk hun eerste boek is. ●